Compressive sensing
a summary of reconstruction algorithms

Master Thesis

Author(s):
Pope, Graeme

Publication date:
2009

Permanent link:
https://doi.org/10.3929/ethz-a-005763272

Rights / license:
In Copyright - Non-Commercial Use Permitted

ETH

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Graeme Pope

Compressive Sensing
A Summary of Reconstruction Algorithms

Abstract

This thesis provides a summary of reconstruction algorithms for compressive sensing.

In addition, we present a new algorithm (the Modified Frame Reconstruction or MFR algorithm) for signal reconstruction in compressive sensing. This algorithm generalises previous iterative hard thresholding algorithms. We prove conditions for successful reconstruction of the original data signal. In addition we show that by overestimating the sparsity of the data signal for our new algorithm, the success rate of the algorithm can be increased.

We also give two modifications to this algorithm: the incorporation of a least squares step and an adaptive method for choosing the step-length. We prove that both algorithms converge to the correct solution under conditions similar original un-modified algorithm. Empirical evidence shows that these modifications dramatically increases both the success rate and the rate of convergence of the modified algorithms in comparison to the un-modified algorithm. Further the improvement is large, up to two orders of magnitude faster. Simulations show that in some cases the modified algorithm outperforms existing compressed sensing reconstruction methods.

Contents

List of Algorithms

Part I

Introduction

Notation

In this document, unless otherwise said, we will denote scalars in lower case, e.g. x, vectors $\mathbf{x}$ in bold font and matrices X in capitals. We will use $\mathbb{R}$ to represent the real numbers and $\mathbb{C}$ to represent the complex numbers. When we write $X \in \mathbb{R}^{m \times n}$ we will take this to mean that X is an $m \times n$ matrix with entries from the real numbers. Similarly $X \in \mathbb{C}^{m \times n}$ means X is a complex valued $m \times n$ matrix. We will use $\cdot^{\mathsf{T}}$ and $\cdot^{*}$ to represent real and complex conjugate transpose for a matrix. A complete listing of the notation used can be seen in Table 1.

We now present a few basic properties of matrices, norms and distributions.

0.1 Matrices & Vectors

Recall the following facts about matrices, eigenvalues and singular values. Let $A \in \mathbb{C}^{m \times n}$ be a matrix with singular values $\sigma_1, \ldots, \sigma_t$, ordered so that $0 \leqslant |\sigma_1| \leqslant |\sigma_2| \leqslant \cdots \leqslant |\sigma_t|$ where $t = \min\{m, n\}$. Now let $\lambda_1, \ldots, \lambda_n$ be the ordered eigenvalues of $A^{\mathsf{T}} A$, then these eigenvalues satisfy $\lambda_i = |\sigma_i|^2$, for $i = 1, \ldots, t$. Similarly for the eigenvalues of AA^{T}. Let $B \in \mathbb{C}^{n \times n}$ be a square matrix. Then the Rayleigh quotient

$$\rho_B(\mathbf{x}) \triangleq \frac{\mathbf{x}^{\mathsf{T}} B \mathbf{x}}{\mathbf{x}^{\mathsf{T}} \mathbf{x}}, \tag{1}$$

is bounded by the maximum and minimum eigenvalues of B, i.e. $\lambda_{\min}(B) \leqslant \rho_B(\mathbf{x}) \leqslant \lambda_{\max}(B)$ for all $\mathbf{x} \in \mathbb{C}^n$.

We will also frequently refer to the Moore-Penrose pseudo-inverse $A^{\dagger}$ of a matrix $A \in \mathbb{C}^{n \times n}$. This is defined to be

$$A^{\dagger} \triangleq (A^* A)^{-1} A^*, \tag{2}$$

but if this does not exist, for instance if neither the rows nor columns are linearly independent,

it can be defined via the limiting process

$$A^\dagger \triangleq \lim_{\delta \to 0} \left(A^*A + \delta \mathbf{I}\right)^{-1} A^*, \tag{3}$$

which exists, even if the inverse of A^*A does not. The pseudo-inverse is of interest because it gives the solution to the least-squares minimisation problem

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x} \in \mathbb{C}^n} \|\mathbf{y} - A\mathbf{x}\|_2, \tag{4}$$

given $\mathbf{y} \in \mathbb{C}^m$ and $A \in \mathbb{C}^{m \times n}$. The solution, $\hat{\mathbf{x}}$, is given by $\hat{\mathbf{x}} = A^\dagger A^* \mathbf{y}$.

For a vector $\mathbf{x} \in \mathbb{R}^n$ we will write $\mathbf{x}^s$ to denote the best s-sparse approximation to $\mathbf{x}$, given by the s-largest (in magnitude) components. We will show later that this is a minimiser of

$$\|\mathbf{x} - \mathbf{x}^s\|_p \tag{5}$$

for any $1 \leqslant p < \infty$. The ℓ_p norm is defined in the next section.

Now let $\Lambda \subset \{1, 2, \ldots, n\}$ be a set of indices and let $k = |\Lambda|$. Let $\mathbf{x} \in \mathbb{R}^n$ be a vector. Then we write $\mathbf{x}_\Lambda \in \mathbb{R}^n$ to denote the vector that agrees with $\mathbf{x}$ on the components given by Λ and 0 elsewhere, i.e.

$$\mathbf{x}_\Lambda \triangleq \begin{cases} (\mathbf{x}_\Lambda)_i = x_i & \text{if } i \in \Lambda \\ (\mathbf{x}_\Lambda)_i = 0 & \text{otherwise.} \end{cases} \tag{6}$$

For a matrix $X \in \mathbb{R}^{m \times n}$ or $X \in \mathbb{C}^{m \times n}$ we take X_Λ to mean the $m \times k$ matrix with columns specified by Λ.

0.2 Norms

Throughout this thesis we will make use of a number of norms on Hilbert spaces.

Definition 0.1. *A Hilbert space $\mathscr{H}$ is a real or complex inner product space that is complete under the norm defined by the inner product. For any $\mathbf{x} \in \mathscr{H}$, we write*

$$\|\mathbf{x}\|_{\mathscr{H}} \triangleq \langle \mathbf{x}, \mathbf{x} \rangle_{\mathscr{H}}, \tag{7}$$

where $\langle \cdot, \cdot \rangle_{\mathscr{H}} : \mathscr{H} \times \mathscr{H} \longrightarrow \mathbb{R}$ is the inner product.

If it is clear from the context we will occasionally drop the subscript. Examples of Hilbert spaces include $\mathbb{R}^n$ with the Euclidean norm.

Throughout this thesis we will talk about ℓ_p norms which give rise to Hilbert spaces over the real and complex numbers. For $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ or $\mathbf{x}, \mathbf{y} \in \mathbb{C}^n$

$$\langle \mathbf{x}, \mathbf{y} \rangle_p \triangleq \left(\sum_{i=1}^n |x_i \overline{y_i}|^p \right)^{\frac{1}{p}}, \tag{8}$$

defines an inner product which gives rise to the norm

$$\|\mathbf{x}\|_p \triangleq \left(\sum_{i=1}^{n} |x_i|^p \right)^{\frac{1}{p}}. \tag{9}$$

on $\mathbb{R}^n$ or $\mathbb{C}^n$ respectively. The space $\mathbb{R}^n$ or $\mathbb{C}^n$ is complete in this norm and hence a Hilbert space. We are particular interested by the case where $p = 2$, which is the standard Euclidean norm and the case $p = 1$. We will also use the ℓ_0 quasi-norm given by

$$\|\mathbf{x}\|_0 \triangleq |\{x_i \neq 0 \colon i = 1, \ldots, n\}|, \tag{10}$$

which is the number of non-zero components. This is a quasi-norm as it fails to obey the triangle inequality. We are also interested in the ℓ_∞ norm which is defined to be the maximum component in magnitude, i.e.

$$\|\mathbf{x}\|_\infty \triangleq \max_{i=1,\ldots,n} |x_i|. \tag{11}$$

From these we can define the *induced* or *operator norm* on the space of real or complex matrices $\mathbb{F}^{m \times n}$ for $\mathbb{F} = \mathbb{R}$ or $\mathbb{F} = \mathbb{C}$. For a matrix $A \in \mathbb{F}^{m \times n}$ we define the p-norm $\|\cdot\|_p : \mathbb{F}^{m \times n} \to \mathbb{R}$, for $1 \leqslant p < \infty$ to be

$$\|A\|_p \triangleq \max_{\mathbf{x} \in \mathbb{R}^n \colon \|\mathbf{x}\|_p = 1} \|A\mathbf{x}\|_p. \tag{12}$$

For the special case of $p = 2$ this gives the *spectral norm*, equal to the largest singular value of A.

0.3 Distributions and Probabilities

We will use $\mathcal{N}(\mu, \sigma^2)$ to represent the Gaussian or Normal distribution with mean μ and variance σ^2. We write $\chi^2(k)$ to denote the χ^2-distribution with k degrees of freedom, which has mean k and variance $2k$.

If X_i are independent $\mathcal{N}(0, 1)$ random variables, then

$$Q = \sum_{i=1}^{m} X_i^2, \tag{13}$$

is distributed according to the χ^2 distribution with m degrees of freedom.

Type	Symbol	Description
Spaces	$\mathbb{R}, \mathbb{C}$	The set of real and complex numbers.
	$\mathbb{F}$	Any field.
	$\mathscr{H}$	A Hilbert space.
Vectors	$x \in \mathbb{F}$	A scalar in the field $\mathbb{F}$.
	$\mathbf{x} \in \mathbb{F}^n$	A column vector with n entries in $\mathbb{F}$.
	x_i	The ith entry of the vector $\mathbf{x}$.
	$\mathbf{0}, \mathbf{1}$	The vectors of all zeroes and all ones, respectively.
	$\mathrm{supp}(\mathbf{x})$	The support of $\mathbf{x}$ which is the position of the non-zero components of $\mathbf{x}$.
	$\mathbf{x}^s$	The best s-sparse approximation to $\mathbf{x}$.
	$\mathbf{x}_\Lambda \in \mathbb{R}^n$	For $\mathbf{x} \in \mathbb{R}^n$ the vector that agrees with $\mathbf{x}$ on Λ and 0 elsewhere.
	$\mathbf{x}^{(k)}$	The k-th iterate of the vector $\mathbf{x}$ in some algorithm.
	$\Sigma_\Gamma \subset \mathbb{R}^n$	The set of all vectors $\mathbf{x} \in \mathbb{R}^n$ with support contained in Γ.
Matrices	$A \in \mathbb{F}^{m \times n}$	An $m \times n$ matrix with entries from $\mathbb{F}$.
	A_{ij}	The (i, j)-th entry of the matrix A.
	$A^*, A^\mathsf{T}, A^\dagger$	Complex conjugate, real transpose and Moore-Penrose pseudo-inverse of A.
	A_Λ	The matrix formed by taking the columns of A specified by Λ.
	$\ker(A)$	The kernel or null-space of the matrix A, $\mathbf{x} \in \ker(A) \iff A\mathbf{x} = \mathbf{0}$.
	$\sigma(A), \sigma_i(A)$	Singular values of A and the i-th largest singular value.
	$\lambda(A), \lambda_i(A)$	Eigenvalues of A and the i-th largest eigenvalue.
	$\mathbf{I}$	The $n \times n$ identity matrix.
	$\|A\|_2$	The induced ℓ_2 norm of a matrix: $\|A\|_2 \triangleq \max_{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x}\|_2 = 1} \|A\mathbf{x}\|_2 = \sigma_{\max}(A)$.
Norms	$|\mathcal{S}|$	The number of elements in the set $\mathcal{S}$.
	$\|\mathbf{x}\|_0$	ℓ_0 "norm": $\|\mathbf{x}\|_0 \triangleq |\{x_i \neq 0\}|$
	$\|\mathbf{x}\|_1$	ℓ_1 norm: $\|\mathbf{x}\|_1 \triangleq \sum_{i=1}^n |x_i|$
	$\|\mathbf{x}\|_2$	ℓ_2 norm: $\|\mathbf{x}\|_2 \triangleq \sqrt{\sum_{i=1}^n x_i^2}$
	$\|\mathbf{x}\|_p$	ℓ_p norm: $\|\mathbf{x}\|_p \triangleq \left(\sum_{i=1}^n |x_i|^p\right)^{1/p}$
	$\|\mathbf{x}\|_\infty$	ℓ_∞ norm: $\|\mathbf{x}\|_\infty \triangleq \max_{i=1,\dots,n} |x_i|$
Distributions	$\mathbb{P}[\mathcal{E}]$	The probability of the event $\mathcal{E}$.
	$\mathbb{E}[X], \mathbb{V}[X]$	Expected value and variance of the random variable X.
	$\mathcal{N}(\mu, \sigma^2)$	The normal/Gaussian distribution with mean μ and variance σ^2.
	$\chi^2(m)$	The χ^2 distribution with m degrees of freedom.

Table 1: Notation and symbols used.

1

Introduction

Compressive sensing is a radical new way of sampling signals at a sub-Nyquist rate. The Shannon/Nyquist sampling theorem states that an analogue signal can be reconstructed perfectly from its samples, if it was sampled at a rate at least twice the highest frequency present in the signal (Nyquist 1928; Shannon 1949). This rate is known as the *Nyquist* or *Shannon* rate of that signal, and for many signals, such as audio or images, the Nyquist rate can be very high. This may result in acquiring a very large number of samples, which must be compressed in order to store or transmit them, as well as placing a high requirement on the equipment needed to sample the signal. *Compressive Sensing* (also referred to as compressed sensing or CS) is a recently introduced method that can reduce the number of measurements required; in some ways it can be regarded as automatically compressing the signal. Compressive sensing is a technique that enables us to fully reconstruct particular classes of signals if the original signal is sampled at a rate well *below* the Nyquist rate.

In particular, compressive sensing works with sparse signals. In many applications the signal of interest is primarily zero, that is, the signal has a representation in some pre-determined basis in which most of the coefficients are 0. Traditionally measurement techniques heavily over sample the signal. Consider the scenario where we randomly draw samples from a sparse signal, so the probability of sampling at an "interesting" data point is equal to the sparsity fraction. Compressive sensing avoids excessive oversampling by using linear sampling operators – a combination of sampling and compression, giving rise to its name.

One of the original breakthroughs in compressive sensing by Candès, Romberg, Tao in (Candès, Romberg, and Tao 2006a; Candès, Romberg, and Tao 2006b; Candès and Tao 2006; Candès and Tao 2005) and Donoho in (Donoho 2006) was to show that linear programming methods can be used to efficiently reconstruct the data signal with high accuracy. Since then many alternative methods have been proposed as a faster or superior (terms of reconstruction rate) alternative to these linear programming algorithms. One approach is to use matching pursuit techniques, originally proposed in (Mallat and Zhang 1993), variations have been proposed such as OMP or orthogonal matching pursuit (Tropp and Gilbert 2007), Stagewise orthogonal matching pur-

suit (StOMP) (Donoho, Tsaig, Drori, and Starck 2007), Compressive sampling matching pursuit (CoSaMP) (Needell and Tropp 2008) and gradient pursuit algorithms (Blumensath and Davies 2008; Blumensath and Davies 2008a; Blumensath and Davies 2008b). Also proposed has been a suite of thresholding based algorithms, either hard thresholding (Blumensath and Davies 2008) or soft thresholding (Daubechies, Defrise, and Mol 2004).

In this thesis we propose a combination of these two techniques, combining hard thresholding with matching pursuit.

In addition to this, work has also been done on model based compressive sensing in (Baraniuk, Cevher, Duarte, and Hegde 2008), which can be applied to many of the algorithms above. Most of the aforementioned algorithms make use of a pruning step which takes a solution and forces it to be sparse, by removing all but the s-largest (in magnitude) components (where s is the sparsity of the signal), which is the best s sparse approximation under any ℓ_p norm for $1 \leqslant p < \infty$. Model-based compressive sensing proposes using a model based algorithm to perform this; that is to choose the sparse signal that is not necessarily best under an ℓ_p norm, but that best fits the signal model. Such a modification would also be applicable to our algorithm.

We refer to our algorithm as the "Modified Frame Reconstruction algorithm" or MFR since it is from the frame reconstruction algorithm that we drew our inspiration. It is however also a generalisation of the iterative hard thresholding (IHT) algorithm first mentioned in (Blumensath and Davies 2008) and could also be correctly referred to as such. However throughout this paper we will use MFR to refer to our algorithm and IHT to refer to the original iterative hard thresholding algorithm.

1.1   Contribution

We make the following contributions with this thesis.

▶ We present a new algorithm that generalises the IHT algorithm.

▶ We prove convergence results for this new algorithm.

▶ We demonstrate two modifications to this algorithm, that increases its performance and prove convergence for both of these cases.

  ◆ We show that adding a least squares minimisation step increases the convergence rate and reconstruction rate.

  ◆ We investigate a way of choosing an adaptive step-length that also increases the rate of convergence and rate of success.

▶ We give new results showing what happens when we do not correctly estimate the sparsity of the signal we are trying to reconstruct.

▶ We give empirical evidence that the algorithm can perform *better* if we underestimate the sparsity, that is, we think the signal we are trying to reconstruct is less sparse than it really is.

These results are currently being written up for publication.

1.2   Organisation

In this thesis we will present the original compressive sensing method and we will also develop and explore a number of other techniques for restoring the original signal, known as reconstruction techniques.

The work in Chapters 6-8 is based on my research and, unless otherwise stated, is my own. Also, unless otherwise stated, if a lemma or theorem is cited and the proof provided, the proof provided is heavily based on the proof in the original paper.

In Chapter 2 we present a background to the theory required. In particular we give a summary of the original compressive sensing theory using ℓ_1 minimisation as the reconstruction algorithm. This is followed by a chapter on the restricted isometry property, including a number of lemmas that we use later. Chapter 4 gives a background to frames, which served as the inspiration to out algorithm.

The following chapter, Chapter 5, has a discussion of alternatives to solving the ℓ_1 minimisation problem for compressive sensing. We give a number of options ranging from greedy algorithms to gradient pursuit algorithms.

In Chapter 6 we present our own reconstruction algorithm for compressive sensing, including an analysis of its performance. We show both theoretically and empirically via simulations how the algorithm performs. We also give two modifications to the algorithm that provide both faster convergence and superior reconstruction performance. For these two modifications we again proved convergence under similar conditions to that of the original algorithm. We then compare the algorithm to previous algorithms, highlighting some of the important differences.

The final two chapters, Chapters 7 and 8, we compare the algorithms discussed, including some simulated performance comparisons and a discussion on the Restricted Isometry Property, how this applies to the theorems we use and a means of estimating the RIP for a given matrix.

1.3   Acknowledgements

I would like to thank my supervisor, Professor Helmut Bölcskei, for introducing me to this area and supervising me during my Masters.

Part II

Background

2

Compressed Sensing

2.1 Goal of Compressed Sensing

Let $f(t)$ be a signal with bandwidth $B > 0$. Then the Shannon-Nyquist sampling theorem says that to perfectly reconstruct this signal from its samples we need to sample it at the Nyquist rate, equal to twice the bandwidth of the signal, i.e. $2B$, (Shannon 1949; Nyquist 1928). We will show that by using compressed sensing techniques we can use a significantly lower sampling rate, but still effect perfect reconstruction for a large class of signals.

Consider a real valued, finite length, one-dimensional, discrete time signal $\mathbf{x} \in \mathbb{R}^n$ (which we regard as an $n \times 1$ column vector). A higher dimensional signal, such as an image, we treat as a one dimensional vector, by writing down its elements in some order. The claim of compressive sensing is that from m measurements where $m \ll n$, we can often perfectly reconstruct the original signal $\mathbf{x}$ where the measurements are not chosen in an adaptive manner.

Now let $\{\psi_i\}_{i=1}^n$ be a set of n orthonormal basis vectors for the space $\mathbb{R}^n$. Let $\Psi \in \mathbb{R}^{n \times n}$ be an orthonormal matrix where the i-th column is the i-th basis vector ψ_i. Then we can express any signal $\mathbf{x} \in \mathbb{R}^n$ as a linear combination of these basis vectors by

$$\mathbf{x} = \sum_{i=1}^n z_i \psi_i \qquad \text{or} \qquad \mathbf{x} = \Psi \mathbf{z}, \tag{2.1}$$

where $\mathbf{z} \in \mathbb{R}^n$ is the vector of inner products $z_i = \langle \mathbf{x}, \psi_i \rangle$. The two vectors $\mathbf{x}$ and $\mathbf{z}$ are equivalent representations of the same signal. Typically we say that $\mathbf{x}$ is in the *time domain* (if it is a time dependent signal, such as audio) or in the *spatial domain* if it is a spatially dependent signal, such as an image and we say that $\mathbf{z}$ is in the Ψ *domain*.

We measure the signal $\mathbf{x}$ by sampling it with respect to a a *measurement matrix* $\Phi \in \mathbb{R}^{m \times n}$. Let Φ have rows ϕ_i for $1 \leqslant i \leqslant m$. Each observation y_i corresponds to the inner product $y_i = \langle \phi_i, \mathbf{x} \rangle$. Writing this in matrix-vector notation we get $\mathbf{y} = \Phi \mathbf{x}$. It is clear that if $m \geqslant n$ and the rows of Φ span $\mathbb{R}^n$ we can perfectly reconstruct (up to numerical precision) the vector $\mathbf{x}$ from its

observations **y**. Substituting $\mathbf{x} = \Psi\mathbf{z}$ into Equation (2.1) we get

$$\mathbf{y} = \Phi\mathbf{x} = \Phi\Psi\mathbf{z} = \Theta\mathbf{z}, \tag{2.2}$$

where $\Theta = \Phi\Psi$.

Compressive sensing asks, what can we do if $m \ll n$? How well can we reconstruct the signal **x** (or **z**)? What assumptions do we require on the the data or the matrix Φ for good reconstruction?

In general we assume that **z** is sparse, that is, it is a linear combination of only $k \ll n$ basis vectors. We say that a vector **z** *is k-sparse with respect to the basis* Ψ, or simply that **z** *is k-sparse*, if it is clear with what basis this is with respect to. Note that solving Equation (2.2) for **z** is equivalent to solving for **x** as Ψ is a known pre-determined basis.

The goal of compressive sensing is to design the matrix Φ and a reconstruction algorithm so that for *k*-sparse signals we require only a "small" number of measurements, i.e. $m \approx k$ or slightly larger.

This does not violate the Shannon-Nyquist sampling theorem as we are not able to reconstruct *all* signals, only sparse signals.

2.2   Measurement Matrix

We first explain how to design the matrix Φ. The ultimate goal is to have some matrix Φ which does not destroy any information contained in the original signal **x**. However since $\Phi \in \mathbb{R}^{m \times n}$ and $m < n$ this is not possible in general as Equation (2.2) is under-determined, making the problem of solving for **z** or **x** ill-posed.

By restricting ourselves to *k*-sparse signals we can do significantly better. If the position of the *k* non-zero entries of **z** were known a priori, i.e. if we were given $\Lambda = \{i | z_i \neq 0\}$, we could form the $m \times k$ matrix Θ_Λ where $m \geqslant k$ and solve the least squares problem restricted to the non-zero positions of **x**. A necessary and sufficient condition for this $m \times k$ system of equations to be well conditioned is that for any *k*-sparse vector $\mathbf{v} \in \mathbb{R}^n$ we have

$$1 - \varepsilon \leqslant \frac{\|\Theta\mathbf{v}\|_2}{\|\mathbf{v}\|_2} \leqslant 1 + \varepsilon, \tag{2.3}$$

for some $\varepsilon > 0$, that is the matrix Θ must almost preserve the lengths of these *k*-sparse vectors. It is unlikely that the positions of the non-zero elements are known a priori, but one can show that a sufficient condition for a stable inverse for *k*-sparse signals is for Θ to satisfy Equation (2.3) and the *restricted isometry property* of order $3k$ to hold (Candès, Romberg, and Tao 2006a). We define the restricted isometry property now.

Definition 2.1. *For each integer* $k = 1, 2, \ldots$ *define the isometry constant* $\delta_k \geqslant 0$ *of a matrix* Φ *as the smallest number such that*

$$1 - \delta_k \leqslant \frac{\|\Phi\mathbf{x}\|_2^2}{\|\mathbf{x}\|_2^2} \leqslant 1 + \delta_k, \tag{2.4}$$

holds for all k-sparse vectors **x**. *We say that a matrix* Φ *has the* restricted isometry property *(RIP) of order k if* δ_k *is not too close to 1.*

The restricted isometry constants give a measure of how much they can change the length of a k-sparse vector. They also relate to the kernel of the matrix: suppose for instance that $\delta_k < 1$ for some integer k. This implies that that there are no k-sparse vectors in the kernel of Φ. Assume for instance that $\mathbf{x}$ is k-sparse and we have $\Phi\mathbf{x} = \mathbf{0}$, this implies $\|\Phi\mathbf{x}\|_2^2 = 0$ and hence $\delta_k \geqslant 1$, which is a contradiction.

Because of the similarities with eigenvalues we will sometimes refer to the values δ_k as the *sparse eigenvalues*. Although not strictly eigenvalues, they inspire the following definition.

Definition 2.2. *Let $\Phi \in \mathbb{R}^{m \times n}$ be a matrix and $k \leqslant \min\{m, n\}$ be an integer. Then the upper and lower k-sparse singular values $\sigma_{\min}^k$ and $\sigma_{\max}^k$ are given by*

$$\sigma_{\min}^k \triangleq \min_{\mathbf{x}} \frac{\|\Phi\mathbf{x}\|_2}{\|\mathbf{x}\|_2} \tag{2.5a}$$

$$\sigma_{\max}^k \triangleq \max_{\mathbf{x}} \frac{\|\Phi\mathbf{x}\|_2}{\|\mathbf{x}\|_2}, \tag{2.5b}$$

where $\mathbf{x}$ is k-sparse. And the upper and lower k-sparse eigenvalues values $\lambda_{\min}^k$ and $\lambda_{\max}^k$ are given by

$$\lambda_{\min}^k \triangleq \min_{\mathbf{x}} \frac{\left\|\mathbf{x}^\mathsf{T}\Phi^\mathsf{T}\Phi\mathbf{x}\right\|_2}{\left\|\mathbf{x}^\mathsf{T}\mathbf{x}\right\|_2} = \left|\sigma_{\min}^k\right|^2 \tag{2.6a}$$

$$\lambda_{\max}^k \triangleq \max_{\mathbf{x}} \frac{\left\|\mathbf{x}^\mathsf{T}\Phi^\mathsf{T}\Phi\mathbf{x}\right\|_2}{\left\|\mathbf{x}^\mathsf{T}\mathbf{x}\right\|_2} = \left|\sigma_{\max}^k\right|^2, \tag{2.6b}$$

where $\mathbf{x}$ is k-sparse.

These definitions are inspired by the fact that these values are given by the maximum and minimum singular/eigen-values over all submatrices with k columns, for example we have

$$\lambda_{\max}^k = \max_{\Gamma : |\Gamma| \leqslant k} \lambda_{\max}\left(\Phi_\Gamma^\mathsf{T}\Phi_\Gamma\right), \tag{2.7}$$

where Γ is an index set of k columns. We have the following relationship between the RIP constants and the sparse eigenvalues

$$\delta_k = \max\left\{1 - \lambda_{\min}^k, \ \lambda_{\max}^k - 1\right\}. \tag{2.8}$$

We also define a weaker version of the restricted isometry property, the *null space property*, which we will use later.

Definition 2.3. *A matrix $\Phi \in \mathbb{R}^{m \times n}$ obeys the* null space property *(NSP) of order k for $\gamma > 0$ if*

$$\|\mathbf{n}_\Lambda\|_1 \leqslant \gamma \|\mathbf{n}_{\Lambda^c}\|_1, \tag{2.9}$$

for all index sets Λ of cardinality less than or equal to k, and for all $\mathbf{n} \in \ker(\Phi)$.

The RIP and NSP are closely linked, as shown in (Cohen, Dahmen, and DeVore 2008).

The important question is how can one create a matrix Φ, given the basis Ψ, so that Φ has the RIP of high order? Verifying that any given matrix has this property is computationally

intensive and involves checking all $\binom{n}{k}$ submatrices with k columns of Θ. In Chapter 8 we shall give a Monte Carlo method for obtaining a lower bound for the RIP constants.

One approach to obtaining a matrix Φ with the RIP of high order is to use random matrices. The $m \times n$ matrices generated according to the following rules:

- ▶ form Φ by sampling n column vectors uniformly on the unit sphere in $\mathbb{R}^m$,

- ▶ let the entries of Φ be i.i.d. normal with mean 0 and variance $\frac{1}{m}$, or

- ▶ let the entries Φ by i.i.d. symmetric Bernoulli distributed, e.g. $\Phi_{ij} = \pm\frac{1}{\sqrt{m}}$ with equal probability, or any other subgaussian distribution,

all obey the restricted isometry property of order k provided that

$$m \geqslant C \cdot k \log\left(\frac{n}{k}\right), \tag{2.10}$$

for some constant C. The proofs that these matrices satisfy the RIP of high order can be found in (Baraniuk, Davenport, DeVore, and Wakin 2007) and (Mendelson, Pajor, and Tomczak-Jaegermann 2006). In Section 3.2 we will prove the result for matrices with Gaussian entries.

One can also ensure stability of the measurement matrix by requiring a high degree of *incoherence* between the measurement matrix Φ and the basis matrix Ψ.

Definition 2.4. *The* coherence *between the sensing basis Φ and the representation basis Ψ is defined to be*

$$\mu\left(\Phi, \Psi\right) \triangleq \sqrt{n} \cdot \max_{1 \leqslant j,k \leqslant n} \left|\left\langle \varphi_k, \psi_j \right\rangle\right|. \tag{2.11}$$

For orthonormal Φ and Ψ it follows that $1 \leqslant \mu\left(\Phi, \Psi\right) \leqslant \sqrt{n}$ for any pairs of matrices Φ and Ψ. The coherence effectively measures the largest correlation between any two elements of Φ and Ψ. A small coherence says that the basis vectors cannot sparsely represent the vectors in Φ and vice-versa, (Candès, Romberg, and Tao 2006a; Donoho 2006).

We also make the following definition of *restricted orthogonality constants* from (Candès and Tao 2005).

Definition 2.5. *Let $\Phi \in \mathbb{R}^{m \times n}$ be a matrix. Then define the s, s'-restricted orthogonality constant $\theta_{s,s'} \geqslant 0$ for $s + s' = S$ to be the smallest quantity such that*

$$\left|\left\langle \Phi\mathbf{x}, \Phi\mathbf{x}' \right\rangle\right| \leqslant \theta_{s,s'} \cdot \|\mathbf{x}\|_2 \cdot \|\mathbf{x}'\|_2, \tag{2.12}$$

for all vectors $\mathbf{x}, \mathbf{x}'$ with disjoint support, where $\mathbf{x}$ is s-sparse and $\mathbf{x}'$ is s'-sparse.

We then have the following lemma relating the restricted orthogonality constants and restricted isometry constants.

Lemma 2.6 ((Candès and Tao 2005)). *Let $\Phi \in \mathbb{R}^{m \times n}$ be a matrix with restricted orthogonality and restricted isometry constants $\theta_{s,s'}$ and δ_S, respectively, and where $S = s + s'$. Then*

$$\theta_{s,s'} \leqslant \delta_S \leqslant \theta_{s,s'} + \max\left\{\delta_s, \delta_{s'}\right\}. \tag{2.13}$$

2.3   Classical Reconstruction Algorithm

The previous section gave us requirements for the measurement matrices Φ, so that $\mathbf{x}$ can be fully recovered given the measurements $\mathbf{y} = \Phi\mathbf{x} = \Phi\Psi\mathbf{z}$. As $\mathbf{z}, \mathbf{x} \in \mathbb{R}^n$ and $\mathbf{y} \in \mathbb{R}^m$ for $m < n$ there are infinitely many solutions to this equation, all lying on a translated copy of the hyperplane

$$\ker(\Phi\Psi) = \{\mathbf{z} \in \mathbb{R}^n \colon \Phi\Psi\mathbf{z} = \mathbf{0}\}, \tag{2.14}$$

that is, the kernel of $\Phi\Psi$.

For simplicity, from this point forth we will consider only the problem $\mathbf{y} = \Phi\mathbf{x}$ where $\mathbf{x}$ is sparse. Since Ψ is a known basis, knowledge of $\mathbf{x}$ is equivalent to knowledge of $\mathbf{z}$.

Classically one solves this type of inverse problem by finding the least squares solution to $\mathbf{y} = \Phi\mathbf{x}$, i.e. solving the problem

$$\tilde{\mathbf{x}} = \arg\min \|\hat{\mathbf{x}}\|_2 \qquad \text{subject to} \qquad \Phi\hat{\mathbf{x}} = \mathbf{y}. \tag{2.15}$$

Although this equation has a convenient closed form solution given by $\tilde{\mathbf{x}} = \left(\Phi\Phi^\mathsf{T}\right)^{-1}\Phi^\mathsf{T}\mathbf{y}$, the solution is almost never sparse.

Since the ℓ_2 or least squares minimisation problem does not usually return a sparse vector, alternatives have been sought. One is to directly enforce a sparsity constraint on the solution, that is to solve the minimisation problem

$$\hat{\mathbf{x}} = \arg\min \|\hat{\mathbf{x}}\|_0 \qquad \text{subject to} \qquad \Phi\hat{\mathbf{x}} = \mathbf{y}. \tag{2.16}$$

Lemma 2.7 shows that under certain conditions the solution to this ℓ_0 minimisation problem returns the solution $\mathbf{x}$ where $\mathbf{x}$ is s-sparse and $\mathbf{y} = \Phi\mathbf{x}$.

Lemma 2.7 (Lemma 1.2 of (Candès and Tao 2005)). *Let $\Phi \in \mathbb{R}^{m \times n}$ be a matrix with RIP constant $\delta_{2s} < 1$ and let Γ be an index set with $|\Gamma| \leqslant s$. Let $\mathbf{x} \in \mathbb{R}^n$ be a vector with support Γ and set $\mathbf{y} = \Phi\mathbf{x}$. Then $\mathbf{x}$ is the unique solution to*

$$\arg\min_{\hat{\mathbf{x}} \in \mathbb{R}^n} \|\hat{\mathbf{x}}\|_0 \qquad \textit{subject to} \qquad \mathbf{y} = \Phi\hat{\mathbf{x}}, \tag{2.17}$$

and hence $\mathbf{x}$ can be reconstructed exactly from $\mathbf{y}$.

The proof is simple and we include it here, based on the proof in (Candès and Tao 2005).

Proof. We show that there is a unique vector $\mathbf{x}$ with support of size less than or equal to s and such that $\mathbf{y} = \Phi\mathbf{x}$. Suppose, by way of contradiction, that there exists a second sparse vector $\mathbf{z} \in \mathbb{R}^n$, $\mathbf{z} \neq \mathbf{x}$ with $\mathbf{y} = \Phi\mathbf{x} = \Phi\mathbf{z}$, where $\mathbf{z}$ has support Λ, $|\Lambda| \leqslant s$. This gives $\Phi(\mathbf{z} - \mathbf{x}) = \mathbf{0}$. But $\operatorname{supp}(\mathbf{z} - \mathbf{x}) = \Lambda \cup \Gamma$ and $|\Lambda \cup \Gamma| \leqslant 2s$. Hence trivially $\|\Phi(\mathbf{z} - \mathbf{x})\|_2^2 = 0$. But by the RIP property we have

$$0 < (1 - \delta_{2s}) \|\mathbf{z} - \mathbf{x}\|_2^2 \leqslant \|\Phi(\mathbf{z} - \mathbf{x})\|_2^2 = 0, \tag{2.18}$$

which implies that $\|\mathbf{z} - \mathbf{x}\|_2 = 0$ and hence $\mathbf{z} = \mathbf{x}$, which is a contradiction as $\delta_{2s} < 1$. $\square$

In fact this bound can be shown to be tight. Assume that there exists a $2s$-sparse vector $\mathbf{v} \in \mathbb{R}^n$

such that $\mathbf{v}$ lies in the kernel of Φ, then we have

$$1 - \delta_{2s} \leqslant \|\Phi\mathbf{v}\|_2 = 0, \tag{2.19}$$

and hence $\delta_{2s} \geqslant 1$. Now choose a set $\Gamma \subset \{1, 2, \ldots, n\}$ of size s so that $\Gamma \subset \mathrm{supp}(\mathbf{v})$ and set $\mathbf{x} = -\mathbf{v}_{\Gamma}$ so that $\mathbf{x}$ is s-sparse and is the negative of $\mathbf{v}$ on s components. Then

$$\mathbf{y} = \Phi\mathbf{x} = \Phi\mathbf{x} + \Phi\mathbf{v} = \Phi(\mathbf{x} + \mathbf{v}) = \Phi\mathbf{u}, \tag{2.20}$$

where $\mathbf{u} = \mathbf{x} + \mathbf{v}$ is s-sparse and $\mathbf{u} \neq \mathbf{x}$. Hence there is no unique minimiser to $\|\hat{\mathbf{x}}\|_0$ subject to $\mathbf{y} = \Phi\hat{\mathbf{x}}$ and no algorithm will be able to return the correct solution 100% of the time.

The problem is that solving the ℓ_0 minimisation programme is an NP-complete problem and hence extremely computationally intensive. Although this gives the desired solution, in practice it is not feasible to solve this equation. The crux of compressive sensing is to show that there exist significantly faster algorithms, that with very high probability, solve this problem.

Much of the work in compressed sensing relates to using the ℓ_1 norm as a substitute in the ℓ_0 minimisation problem above. It has been shown in (Candès and Romberg 2007), here Theorem 2.11, that if $\mathbf{x}$ is k-sparse and if the number of measurements m satisfies

$$m \geqslant C \cdot k \cdot \log\left(\frac{n}{\delta}\right), \tag{2.21}$$

then with probability exceeding $1 - \delta$ the solution to the problem

$$\tilde{\mathbf{x}} = \arg\min \|\hat{\mathbf{x}}\|_1 \qquad \text{subject to} \qquad \Phi\hat{\mathbf{x}} = \mathbf{y}, \tag{2.22}$$

is $\tilde{\mathbf{x}} = \mathbf{x}$. Or in the case where there is noise, subject to $\|\Phi\hat{\mathbf{x}} - \mathbf{y}\|_2 \leqslant \varepsilon$, where ε is a tolerance parameter. The ℓ_1 minimisation problem can be solved by rewriting it as a convex optimisation problem

$$(P_1) \qquad \arg\min_{\hat{\mathbf{x}} \in \mathbb{R}^n} \|\mathbf{y} - \Phi\hat{\mathbf{x}}\|_1, \tag{2.23}$$

by Theorem 2.9 from (Candès, Rudelson, Tao, and Vershynin 2005). This reduces to a linear programme

$$(P_1') \qquad \min \mathbf{1}^{\mathsf{T}}\mathbf{t} \quad \text{subject to} \quad -\mathbf{t} \leqslant \mathbf{y} - \Phi\hat{\mathbf{x}} \leqslant \mathbf{t}, \tag{2.24}$$

known as *basis pursuit*, which has computational complexity $\mathcal{O}(n^3)$, which is much easier to solve than the ℓ_0 programme, (Candès, Rudelson, Tao, and Vershynin 2005; Boyd and Vandenberghe 2004; Rockafellar 1970).

A similar minimisation problems that occurs commonly in the literature is to minimise the function

$$F_{\tau}(\mathbf{x}) \triangleq \|\Phi\mathbf{x} - \mathbf{y}\|_2^2 + 2\tau \|\mathbf{x}\|_1, \tag{2.25}$$

for some $\tau > 0$ which is a convex unconstrained optimisation problem. Problems of this form can be seen as a maximum *a posteriori* criterion for estimating $\mathbf{x}$ from the observations $\mathbf{y} = \Phi\mathbf{x} + \mathbf{n}$ where $\mathbf{n}$ is white Gaussian noise and the prior on $\mathbf{x}$ is Laplacian, (Alliney and Ruzinsky 1994). It can be shown that a solution to Equation (2.22) is a minimiser of Equation (2.25) for $\tau = \|\mathbf{x}\|_1$, see (Boyd and Vandenberghe 2004; Rockafellar 1970). This is used as the basis of some algorithms, such as iterative soft-thresholding in (Daubechies, Fornasier, and Loris 2007) which we discuss further in Section 5.2.2.

2.4 Reconstruction Results for ℓ_1 Minimisation

In the literature there are several different theorems saying how well ℓ_1 minimisation works as a way of solving $\mathbf{y} = \Phi\mathbf{x}$ for $\mathbf{x}$.

The main result of (Candès, Rudelson, Tao, and Vershynin 2005) is the following theorem about the (P_1) and (P_1') optimisation programs in Equation (2.23) and (2.24). Consider the scenario where there are corrupted measurements $\mathbf{y} = \Phi\mathbf{x} + \mathbf{e}$ where $\mathbf{e}$ is an error vector and $\Phi \in \mathbb{R}^{m \times n}$. Then let $Y \in \mathbb{R}^{(m-n) \times n}$ be a matrix that annihilates Φ from the left, that is $Y\Phi = \mathbf{0}$. Then $\mathbf{y} = \Phi\mathbf{x} + \mathbf{e}$ implies that

$$Y(\Phi\mathbf{x} + \mathbf{e}) = Y\mathbf{e}. \tag{2.26}$$

Theorem 2.8 (Theorem 1.1 of (Candès, Rudelson, Tao, and Vershynin 2005)). *Let $\Phi \in \mathbb{R}^{m \times n}$ be a matrix with RIP constants δ_{3s} and δ_{4s} such that*

$$\delta_{3s} + 3\delta_{4s} < 2, \tag{2.27}$$

and let $\mathbf{e}$ be an arbitrary vector with support of size less than or equal to s, then the solution to

$$\arg\min_{\hat{\mathbf{e}} \in \mathbb{R}^n} \|\hat{\mathbf{e}}\|_1, \qquad \text{subject to} \qquad Y\hat{\mathbf{e}} = Y\mathbf{e}, \tag{2.28}$$

is unique and equal to $\mathbf{e}$.

It is sufficient then to have $\delta_{4s} < 0.5$ to guarantee convergence. As a consequence we have the following theorem.

Theorem 2.9 (Theorem 1.2 of (Candès, Rudelson, Tao, and Vershynin 2005)). *Let $\mathbf{y} \in \mathbb{R}^m$ be the corrupted measurement signal of $\mathbf{x}$, i.e. $\mathbf{y} = \Phi\mathbf{x} + \mathbf{e}$ where $\Phi \in \mathbb{R}^{m \times n}$ and let $s \geqslant 1$ be a number satisfying the hypothesis of Theorem 2.8. Then if $\mathbf{e}$ also obeys the hypothesis of Theorem 2.8, the solution to*

$$\arg\min_{\hat{\mathbf{x}} \in \mathbb{R}^n} \|\mathbf{y} - \Phi\hat{\mathbf{x}}\|_1, \tag{2.29}$$

is unique and equal to $\mathbf{x}$.

Simply, Theorem 2.9 equates the solution of the two problems

$$\arg\min_{\mathbf{y} = \Phi\hat{\mathbf{x}}} \|\hat{\mathbf{x}}\|_1 \qquad \text{and} \qquad \arg\min_{\hat{\mathbf{x}} \in \mathbb{R}^n} \|\mathbf{y} - \Phi\hat{\mathbf{x}}\|_1, \tag{2.30}$$

under the condition that $\delta_{3s} + 3\delta_{4s} < 2$. A different version of this result is given in (Candès and Tao 2005).

Unfortunately Lemma 2.7 only proves the existence of a unique solution and does not give any way of finding it, beyond a brute force search which is of order $\mathcal{O}(n^s)$ for an s-sparse vector $\mathbf{x} \in \mathbb{R}^n$ and $m \ll n$. The main theorem of (Candès and Tao 2005) gives stronger conditions on the RIP constants so that we can recover the solution via a much faster ℓ_1 minimisation procedure.

Theorem 2.10 (Theorem 1.3 of (Candès and Tao 2005)). *Let s be a positive integer. Suppose $\Phi \in \mathbb{R}^{m \times n}$ has RIP constants such that*

$$\delta_s + \theta_{s,s} + \theta_{s,2s} < 1, \tag{2.31}$$

and let $\mathbf{x} \in \mathbb{R}^n$ *be a vector with support* Γ, $|\Gamma| \leqslant s$ *and set* $\mathbf{y} = \Phi\mathbf{x}$. *Then* $\mathbf{x}$ *is the unique minimiser of*

$$\min_{\hat{\mathbf{x}} \in \mathbb{R}^n} \|\hat{\mathbf{x}}\|_1 \qquad \text{subject to} \qquad \mathbf{y} = \Phi\hat{\mathbf{x}}. \tag{2.32}$$

It follows from Lemma 2.6 that if Equation (2.31) holds, then $\delta_{2s} < 1$.

There is yet another condition for convergence in (Candès and Romberg 2007) which tightens bounds in earlier work, such as (Candès, Romberg, and Tao 2006a; Donoho 2006).

Theorem 2.11 (Theorem 1.1 of (Candès and Romberg 2007)). *Let* U *be an* $n \times n$ *orthogonal matrix and let* $\Phi \in \mathbb{R}^{m \times n}$ *be a matrix formed by taking* m *different rows of* U. *Choose a signal* $\mathbf{x} \in \mathbb{R}^n$ *with support* Γ *of size* s *and a sign sequence* $\mathbf{z}$ *on* Γ *uniformly at random. Suppose that*

$$m \geqslant C_0 \cdot s \cdot \mu^2(U) \cdot \log\left(\frac{n}{\delta}\right), \tag{2.33}$$

and

$$m \geqslant C_1 \cdot \log^2\left(\frac{n}{\delta}\right), \tag{2.34}$$

for some fixed numerical constants C_0 *and* C_1 *and let* $\mu(U) = \max_{k,j}\left|U_{k,j}\right|$ *be the largest entry of* U. *Then with probability exceeding* $1 - \delta$ *every signal* $\hat{\mathbf{x}}$ *supported on* Γ *with signs matching* $\mathbf{z}$ *can be recovered from* $\mathbf{y} = \Phi\mathbf{x}$ *by solving*

$$\arg\min_{\tilde{\mathbf{x}} \in \mathbb{R}^n} \|\tilde{\mathbf{x}}\|_1 \qquad \text{subject to} \qquad \Phi\hat{\mathbf{x}} = \Phi\tilde{\mathbf{x}}. \tag{2.35}$$

In effect this gives a lower bound on the number of measurements required to reconstruct a sparse signal $\mathbf{x}$ from its measurements $\mathbf{y} = \Phi\mathbf{x}$.

Finally we have two results from (Candès and Tao 2006) that say it is possible to reconstruct compressible signals, that is, signals that decay rapidly but are not strictly sparse. Compressible signals are discussed further in Section 2.6, but briefly, we say that a signal $\mathbf{x} \in \mathbb{R}^n$ is compressible if the components decay according to a power law, i.e. if the k-th largest component of $\mathbf{x}$, say $x_{\mathcal{I}(k)}$, satisfies

$$\left|x_{\mathcal{I}(k)}\right| \leqslant C \cdot k^{-\frac{1}{p}}, \tag{2.36}$$

for some constants C, p and for all k.

Theorem 2.12 (Theorem 1.1 of (Candès and Tao 2006)). *Let* $\mathbf{x} \in \mathbb{R}^n$ *be a compressible signal according to Equation (2.36) for some fixed* $0 < p < 1$ *or* $\|\mathbf{x}\|_1 \leqslant R$ *for* $p = 1$, *and let* $\alpha > 0$ *be a sufficiently small number (less than an absolute constant). Assume that* $\Phi \in \mathbb{R}^{m \times n}$ *is a matrix whose rows are orthonormal, chosen uniformly at random and that we are given* $\mathbf{y} = \Phi\mathbf{x}$. *Then with probability 1 the minimiser* $\hat{\mathbf{x}}$ *to*

$$\min_{\mathbf{z} \in \mathbb{R}^n} \|\mathbf{z}\|_1 \qquad \text{subject to} \qquad \mathbf{y} = \Phi\mathbf{z}, \tag{2.37}$$

is unique. Furthermore, with probability at least $1 - \mathcal{O}\left(n^{-\rho/\alpha}\right)$, *we have the approximation*

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_2 \leqslant C_{p,\alpha} \cdot R \cdot \left(\frac{m}{\log n}\right)^{-r}, \qquad r = \frac{1}{p} - \frac{1}{2}. \tag{2.38}$$

Here $C_{p,\alpha}$ *is a fixed constant depending on* p *and* α *but not on anything else. The implicit constant in* $\mathcal{O}\left(n^{-\rho/\alpha}\right)$ *is allowed to depend on* α.

What this results says, is that if we make $\mathcal{O}(m \log n)$ random measurements of a signal $\mathbf{x}$, and then reconstruct an approximation in a way which makes no use of any assumptions on the signal, beyond the fact that it decays according to a power law (of perhaps unknown parameters), we can still obtain an approximation that is equally as good by knowing everything about $\mathbf{x}$ and selecting the m largest components of the vector $\mathbf{x}$. Indeed the amount of "oversampling" required in doing this is only a multiplicative factor of $\mathcal{O}(\log n)$. As such, one cannot, in general, design a set of m measurements that will give a better approximation, by any method.

Theorem 2.13 (Theorem 1.2 of (Candès and Tao 2006))**.** *Let $\Phi \in \mathbb{R}^{m \times n}$ be a measurement process such that the UUP and ERP hold (see Section 3.4.1) with oversampling factors λ_1 and λ_2 respectively. Set $\lambda = \max\{\lambda_1, \lambda_2\}$ and assume $K \geqslant \lambda$. Suppose that $\mathbf{x} \in \mathbb{R}^n$ is a compressible signal according to (2.36) for some fixed $0 < p < 1$ or $\|\mathbf{x}\|_1 \leqslant R$ for $p = 1$ and set $r = \frac{1}{p} - \frac{1}{2}$. Then for any sufficiently small α, any minimiser $\hat{\mathbf{x}}$ to*

$$\min_{\mathbf{z} \in \mathbb{R}^n} \|\mathbf{z}\|_1 \qquad \text{subject to} \qquad \Phi \mathbf{x} = \Phi \mathbf{z}, \tag{2.39}$$

will obey

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_2 \leqslant C_{\rho, \alpha} \cdot R \cdot \left(\frac{K}{\lambda}\right)^{-r}, \tag{2.40}$$

with probability at least $1 - \mathcal{O}\left(n^{-\rho/\alpha}\right)$

In Section 3.4.1 we define the UUP and ERP. Many of the matrices that satisfy the RIP condition of high order also satisfy the UUP and ERP.

2.5 Interpretation

Looking at the two-dimensional case, $\mathbb{R}^2$ we can see that solving the ℓ_1 problem in place of the ℓ_2 problem is much more likely to return a sparse solution. If $n = 2$ we necessarily have $m = 1$, so the set of solutions to $\mathbf{y} = \Phi \mathbf{x}$ is a line. Assume that the solution we are after is 1-sparse, that is, $\mathbf{x}$ lies on one of the two axes. Then minimising $\|\hat{\mathbf{x}}\|_2$ subject to $\mathbf{y} = \Phi \hat{\mathbf{x}}$ returns the solution that is closest to the origin and the only way for this to lie on one of the axes is if the line of solutions is horizontal or vertical, which happens if and only if one of the entries of Φ is 0.

However if we see where the line of solutions touches the smallest ℓ_1 ball, it is highly likely that it will touch at a corner of the ℓ_1 ball giving a sparse solution. Indeed the line of solutions will necessarily touch at a corner, although it is possible that the line touches an entire side of the ℓ_1 ball, which happens if $\Phi = (\pm\lambda \quad \pm\lambda) \in \mathbb{R}^{1 \times 2}$. This is illustrated in Figure 2.1.

We can easily extend this idea to higher dimensions. It is highly unlikely that the hyper-plane of solutions will intersect an ℓ_2 ball on an axis, but since an ℓ_1 ball is "pointy", it will always intersect the ℓ_1 ball at an axis, but this may not be unique. Hence at least one component will be 0, but likely significantly more components.
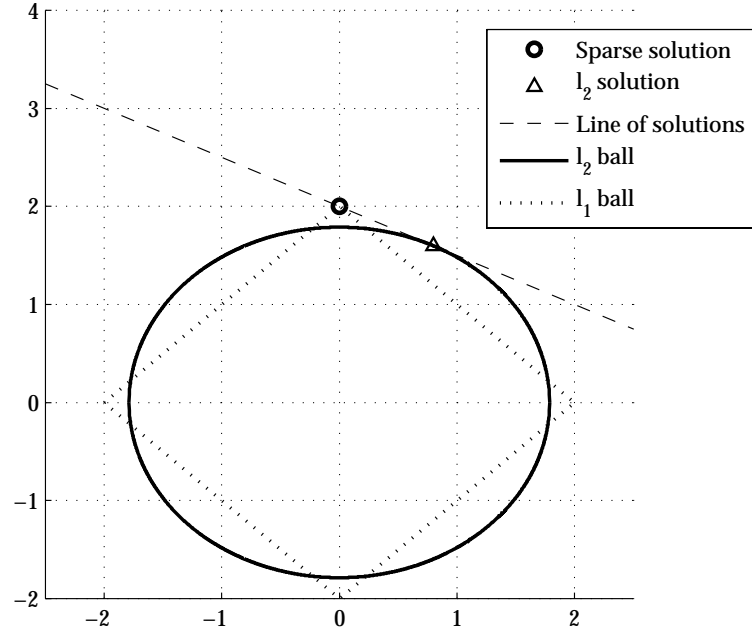
Figure 2.1: Solving the ℓ_1 and ℓ_2 minimisation problem in $\mathbb{R}^2$. Notice how the ℓ_1 ball intersects the line of solutions to give the sparsest solution but the ℓ_2 ball does not.

2.6   Compressible Signals

Many signals occurring both naturally and man-made are not strictly sparse, but can be approximated as such. A signal $\mathbf{x} \in \mathbb{R}^n$ whose coefficients x_i decrease in magnitude according to a power law

$$\left| x_{\mathcal{I}(k)} \right| \leqslant C \cdot k^{-\frac{1}{r}}, \qquad k = 1, \ldots, n \tag{2.41}$$

where $\mathcal{I}(k)$ is the k-th largest component of $\mathbf{x}$ sorted by magnitude from largest to smallest, we call *compressible*. Due to the rapid decay of such signals, these signals can be well approximated by s-sparse signals, keeping just the s largest coefficients of $\mathbf{x}$.

The following lemma states that the best s-term approximation to a vector $\mathbf{x}$ under any ℓ_p norm is the one given by selecting the s largest (in magnitude) components of $\mathbf{x}$.

Lemma 2.14. *Let $\mathbf{x} \in \mathbb{R}^n$ be a signal and let Λ_s the indices of the s largest components, sorted lexicographically if necessary.. Then for any ℓ_p norm, $1 \leqslant p < \infty$ the best s-sparse approximation $\mathbf{x}^s$ is given by*

$$\mathbf{x}^s = \mathbf{x}_{\Lambda_s} = \begin{cases} x_i & \textit{if } i \in \Lambda_s \\ 0 & \textit{else.} \end{cases} \tag{2.42}$$

That is

$$\mathbf{x}^s = \arg \min_{\hat{\mathbf{x}} \in \Sigma_s} \left\| \mathbf{x} - \hat{\mathbf{x}} \right\|_p. \tag{2.43}$$

Proof. Without loss of generality assume that $\mathbf{x} \in \mathbb{R}^n$ is such that $|x_i| \geqslant |x_{i+1}|$ for $1 \leqslant i < n$. Fix p and let $\mathbf{x}^*$ be a solution to Equation (2.43). Then for any $x_i^* \neq 0$ we must have $x_i^* = x_i$.

By assumption the indices of the j-largest components (ties broken lexicographically) are $\Lambda_j = \{1, \ldots, j\}$ and as shorthand we will write $\mathbf{x}^j = \mathbf{x}_{\Lambda_j}$.

Assume that the lemma is false and let $k \in \mathbb{N}$ be the smallest k such that for all best k-sparse approximation, $\hat{\mathbf{x}}$, to $\mathbf{x}$ under the ℓ_p-norm, we have $\mathrm{supp}(\hat{\mathbf{x}}) \neq \Lambda_k$. Let $\mathbf{x}^*$ be such a k-sparse minimiser of (2.43) with support Γ and with the indices smallest, i.e. if $x_i^* = x_{i-1}^*$ then $i \in \Gamma$ implies that $(i-1) \in \Gamma$. This implies that there exists $t > k$ such that $t \in \Gamma$ and $|x_t^*| < |x_k^*|$ (note the strict inequality), since we assume that the support of $\mathbf{x}^*$ is not equal to the first k elements.

So by assumption we have

$$\|\mathbf{x} - \mathbf{x}^*\|_p \leqslant \|\mathbf{x} - \hat{\mathbf{x}}\|_p \qquad \forall\, \hat{\mathbf{x}} \in \Sigma_k \tag{2.44a}$$

$$\text{and} \qquad \left\|\mathbf{x} - \mathbf{x}^j\right\|_p \leqslant \|\mathbf{x} - \hat{\mathbf{x}}\|_p \qquad \forall\, \hat{\mathbf{x}} \in \Sigma_j,\, j < k. \tag{2.44b}$$

In particular

$$\|\mathbf{x} - \mathbf{x}^*\|_p^p = \sum_{\gamma \in \Gamma^c} |x_\gamma|^p \leqslant \sum_{i=k+1}^n |x_i|^p = \left\|\mathbf{x} - \mathbf{x}^k\right\|_p^p \tag{2.45a}$$

$$\Rightarrow \quad \sum_{\gamma \in \Gamma^c} |x_\gamma|^p + |x_t|^p \leqslant \sum_{i=k+1}^n |x_i|^p + |x_t|^p. \tag{2.45b}$$

Set $\Gamma' = \Gamma \setminus \{t\}$ then $\mathbf{x}_{\Gamma'}$ is a $(k-1)$-sparse approximation to $\mathbf{x}$, giving us

$$\left\|\mathbf{x} - \mathbf{x}^{k-1}\right\|_p^p \leqslant \|\mathbf{x} - \mathbf{x}_{\Gamma'}\|_p^p = \sum_{\gamma \in \Gamma^c} |x_\gamma|^p + |x_t|^p \leqslant \sum_{i=k+1}^n |x_i|^p + |x_t|^p, \tag{2.46}$$

where $\mathbf{x}^{k-1}$ is the best $(k-1)$-sparse approximation to $\mathbf{x}$ and $\mathrm{supp}(\mathbf{x}^{k-1}) = \{1, \ldots, k-1\}$. Hence

$$\sum_{i=k}^n |x_i|^p \leqslant \sum_{i=k+1}^n |x_i|^p + |x_t|^p, \tag{2.47}$$

which is true only if $|\mathbf{x}_k| = |\mathbf{x}_t|$, but by assumption this is not true, therefore no such k exists, which is a contradiction. $\qquad\square$

If we let $\sigma_s(\mathbf{x})_p$ denote the error in using the best s-term approximation to $\mathbf{x}$ under the ℓ_p norm, i.e. $\sigma_s(\mathbf{x})_p \triangleq \min_{\hat{\mathbf{x}}} \|\mathbf{x} - \hat{\mathbf{x}}\|_p$ we have that

$$\sigma_s(\mathbf{x})_p \leqslant C(rt)^{-\frac{1}{p}} s^{-t}, \tag{2.48}$$

where $t = \frac{1}{r} - \frac{1}{p}$ and C is the constant from Equation (2.41).

Such approximations form the basis of *transform coding* algorithms, for example the JPEG and JPEG2000 algorithms (Mallat 1999). Such algorithms are not necessarily the most efficient in terms of data collected. To perform transform coding on a signal $\mathbf{x} \in \mathbb{R}^n$, using the transform $\Psi \in \mathbb{R}^{m \times n}$ to get $\mathbf{y} = \Psi \mathbf{x}$, requires acquiring a large number of samples, m, even if the number of samples to keep, s, is significantly smaller than m, which involves discarding $m - s$ of the calculated terms. Compressive sensing asks and answers the question: is there a way of directly acquiring the compressed samples $\mathbf{y}$, that is, so after the acquisition process, we do not discard any data?

3

Restricted Isometry Property

In this chapter we will take a closer look at the Restricted Isometry Property. We will give a condition on a random matrix so that with very high probability it will obey the RIP of some given order. We will then use this to prove that matrices drawn from the Gaussian distribution obey the RIP with high probability.

We will also present a number of lemmas and propositions that make use of the RIP constants, which we will use later in Chapters 5 and 6 when proving results on the convergence of various reconstruction algorithms. Finally we will discuss a number of alternatives to the Restricted Isometry Property that also appear in the literature.

3.1 Proving the Restricted Isometry Property

One of the earliest results of compressed sensing was to show that various randomly generated matrices do in fact obey the RIP of various orders. The results of (Baraniuk, Davenport, DeVore, and Wakin 2007) attempt to give a sufficient condition for various random matrices to obey the RIP.

Let (Ω, ρ) be a probability space and let r be a random variable on Ω. Then given $m, n \in \mathbb{Z}$ we generate random matrices $\Phi \in \mathbb{R}^{m \times n}$ by choosing the entries Φ_{ij} as independent realisations of r, resulting in the random matrices $\Phi(\omega)$, $\omega \in \Omega^{mn}$.

Lemma 3.1 (Lemma 5.1 of (Baraniuk, Davenport, DeVore, and Wakin 2007))**.** *Let (Ω, ρ) be a probability space and let r be a random variable on Ω. Assume that the random matrix $\Phi(\omega) \in \mathbb{R}^{m \times n}$ satisfies*

$$\mathbb{P}\left[\left|\|\Phi(\omega)\mathbf{x}\|_2^2 - \|\mathbf{x}\|_2^2\right| \geqslant \varepsilon \|\mathbf{x}\|_2^2\right] \leqslant 2e^{-mc_0(\varepsilon)}, \qquad 0 < \varepsilon < 1, \tag{3.1}$$

for all $\mathbf{x} \in \mathbb{R}^n$, where $\mathbb{P}$ is the induced probability and $c_0(\varepsilon) > 0$ is a constant depending only on ε. Fix

k < n, then for any set of indices Γ such that |Γ| = k < n and any 0 < δ < 1, we have

$$1 - \delta \leqslant \frac{\|\Phi(\omega)\mathbf{x}\|_2}{\|\mathbf{x}\|_2} \leqslant 1 + \delta, \tag{3.2}$$

for all $\mathbf{x} \in \mathbb{R}^n$ with support Γ, with probability at least

$$1 - 2\left(\frac{12}{\delta}\right)^k e^{-mc_0\left(\frac{\delta}{2}\right)}. \tag{3.3}$$

Proof. As Φ is linear we need only consider the cases where $\|\mathbf{x}\|_2 = 1$. Let $\Sigma_\Gamma \subset \mathbb{R}^n$ be the set of all vectors in $\mathbb{R}^n$ with support Γ. Fix δ and let $\Xi \subset \Sigma_\Gamma$ be a δ/4 covering set of unit vectors for the unit vectors in Σ_Γ, that is, for all $\mathbf{x} \in \Sigma_\Gamma$ where $\|\mathbf{x}\|_2 = 1$ we have

$$\min_{\mathbf{v} \in \Xi} \|\mathbf{x} - \mathbf{v}\|_2 \leqslant \frac{\delta}{4}, \tag{3.4}$$

and $\|\mathbf{v}\|_2 = 1$ for all $\mathbf{v} \in \Xi$. It can be shown that such a set Ξ exists with size at most $(12/\delta)^k$, (Lorentz, Golitschek, and Makovoz 1996, Ch. 13). By application of the union bound, we get that every point in Ξ obeys Equation (3.1) for $\varepsilon = \delta/2$ with probability

$$\left(\frac{12}{\delta}\right)^k 2e^{-mc_0\delta/2}. \tag{3.5}$$

Rearranging the terms in the probability in Equation (3.1), we have

$$1 - \frac{\delta}{2} \leqslant \frac{\|\Phi\mathbf{v}\|_2}{\|\mathbf{v}\|_2} \leqslant 1 + \frac{\delta}{2}, \tag{3.6}$$

with probability exceeding

$$1 - 2\left(\frac{12}{\delta}\right)^k e^{-mc_0\delta/2}, \tag{3.7}$$

for all $\mathbf{v} \in \Xi$. Now define α to be the smallest number such that $\|\Phi\mathbf{x}\|_2 \leqslant (1 + \alpha)\|\mathbf{x}\|_2$ for all $\mathbf{x} \in \Sigma_\Gamma$. We will show that $\alpha \leqslant \delta$. As Ξ is a δ/4 covering set for the unit vectors in Σ_Γ, for any unit vector $\mathbf{x} \in \Sigma_\Gamma$ we have that there exists $\mathbf{v} \in \Xi$ such that $\|\mathbf{x} - \mathbf{v}\|_2 \leqslant \delta/4$. So for all $\mathbf{x} \in \Sigma_\Gamma$ with $\|\mathbf{x}\|_2 = 1$, let $\mathbf{v}_x$ be a vector such that $\|\mathbf{x} - \mathbf{v}_x\|_2 \leqslant \delta/4$. Then

$$\|\Phi\mathbf{x}\|_2 \leqslant \|\Phi\mathbf{v}_x\|_2 + \|\Phi(\mathbf{x} - \mathbf{v}_x)\|_2 \tag{3.8a}$$

$$\leqslant 1 + \frac{\delta}{2} + (1 + \alpha)\frac{\delta}{4}. \tag{3.8b}$$

As α is the smallest number such that $\|\Phi\mathbf{x}\|_2 \leqslant (1 + \alpha)\|\mathbf{x}\|_2$ for all $\mathbf{x} \in \Sigma_\Gamma$, we must have

$$\alpha \leqslant \frac{\delta}{2} + (1 + \alpha)\frac{\delta}{4} \qquad \Rightarrow \qquad \alpha \leqslant \frac{3\delta}{4 - \delta} \leqslant \delta, \tag{3.9}$$

hence proving

$$\frac{\|\Phi(\omega)\mathbf{x}\|_2}{\|\mathbf{x}\|_2} \leqslant 1 + \delta \qquad \forall \mathbf{x} \in \Sigma_\Gamma. \tag{3.10}$$

To show the lower inequality, observe the following

$$\|\mathbf{x}\|_2 \geqslant \|\Phi\mathbf{v}_x\|_2 - \|\Phi(\mathbf{x} - \mathbf{v}_x)\|_2 \tag{3.11a}$$

$$\geqslant 1 - \frac{\delta}{2} - (1 + \delta)\frac{\delta}{4} \tag{3.11b}$$

$$\geqslant 1 - \delta, \tag{3.11c}$$

completing the proof. $\qquad\square$

The following theorem says that if a random matrix obeys the concentration inequality of the previous lemma, it will have the RIP of order s with high probability.

Theorem 3.2 (Theorem 5.2 of (Baraniuk, Davenport, DeVore, and Wakin 2007))**.** *Suppose that m, n and $0 < \delta < 1$ are given. Let Φ satisfy the conditions of Lemma 3.1. Then there exist constants $c_1, c_2 > 0$ depending only on δ such that the restricted isometry property holds for $\Phi(\omega)$ with the prescribed δ and any $k \leqslant \frac{c_1 n}{\log(n/k)}$ with probability at least $1 - e^{-c_2 m}$.*

Proof. From Lemma 3.1 we know that for any index set Γ of size k, the matrix $\Phi(\omega)$ will fail to satisfy the concentration inequality with probability at most

$$2\left(\frac{12}{\delta}\right)^k e^{-mc_0\left(\frac{\delta}{2}\right)}. \tag{3.12}$$

As there are $\binom{n}{k} \leqslant \left(\frac{en}{k}\right)^k$ such index sets, Equation (3.2) will fail to hold with probability at most

$$2\left(\frac{en}{k}\right)^k \left(\frac{12}{\delta}\right)^k e^{-mc_0\left(\frac{\delta}{2}\right)} = 2\exp\left\{-mc_0\left(\frac{\delta}{2}\right) + k\log\left(\frac{en}{k}\right) + k\log\left(\frac{12}{\delta}\right)\right\}. \tag{3.13}$$

Hence if $k \leqslant \frac{c_1 m}{\log(n/k)}$ for each fixed $c_1 > 0$, the exponent above is less than $c_2 \cdot m$ provided that

$$c_2 \leqslant c_0 \cdot \frac{\delta}{2} - c_1 \cdot \left(1 + \frac{1 + \log\left(\frac{12}{\delta}\right)}{\log\left(\frac{n}{k}\right)}\right). \tag{3.14}$$

Therefore we can always choose $c_1 > 0$ sufficiently small ensuring that $c_2 > 0$. So with probability $1 - e^{-mc_2}$ the matrix $\Phi(\omega)$ will satisfy the concentration inequality of Equation (3.1) for each k-sparse $\mathbf{x}$. $\qquad\square$

3.2 Gaussian Measurement Matrix

We can use Theorem 3.2 to prove that a matrix with entries from a Gaussian distribution satisfies the RIP criteria with high probability. Let Φ be an $m \times n$ matrix, $m < n$, with i.i.d. entries $\Phi_{ij} \sim \mathcal{N}\left(0, \frac{1}{m}\right)$. Let ϕ_i for $1 \leqslant i \leqslant m$ be the rows of Φ. To apply the theorem we need to show that for arbitrary $\mathbf{x} \in \mathbb{R}^n$, $\|\Phi\mathbf{x}\|_2^2$ is strongly concentrated about $\|\mathbf{x}\|_2^2$, that is

$$\mathbb{P}\left(\left|\|\Phi\mathbf{x}\|_2^2 - \|\mathbf{x}\|_2^2\right| \geqslant \varepsilon \|\mathbf{x}\|_2^2\right) \leqslant 2e^{-mc_0(\varepsilon)}. \tag{3.15}$$

Define Z_i to be the random variable

$$Z_i \triangleq \phi_i \mathbf{x} = \sum_{j=1}^{n} \phi_{ij} x_j, \tag{3.16}$$

which has distribution $\|\mathbf{x}\|_2 \times \mathcal{N}\left(0, \frac{1}{m}\right)$. For convenience, we consider $\mathbf{x}$ where $\|\mathbf{x}\|_2 = 1$, hence $Z_i \sim \mathcal{N}\left(0, \frac{1}{m}\right)$. Then let Y be the random variable drawn from the χ^2 distribution with m degrees of freedom, i.e. $Y \sim \chi^2(m)$. Observe that

$$\frac{Y}{m} = \|\Phi \mathbf{x}\|_2^2 = \sum_{i=1}^{m} (\phi_i \mathbf{x})^2 = \sum_{i=1}^{m} Z_i^2, \tag{3.17}$$

since the sum of m squared i.i.d. normal random variables is distributed according to the χ^2 distribution. So we need an upper bound for

$$\mathbb{P}\left(\left|\|\Phi \mathbf{x}\|_2^2 - \|\mathbf{x}\|_2\right| \geqslant \varepsilon \|\mathbf{x}\|_2^2\right) = \mathbb{P}\left(\left|\frac{Y}{m} - 1\right| \geqslant \varepsilon\right) \tag{3.18a}$$

$$= \mathbb{P}\left(\left\{Y \geqslant m(1+\varepsilon)\right\} \cup \left\{Y \leqslant m(1-\varepsilon)\right\}\right) \tag{3.18b}$$

$$= \mathbb{P}\left(Y \geqslant m(1+\varepsilon)\right) + \mathbb{P}\left(Y \leqslant m(1-\varepsilon)\right). \tag{3.18c}$$

Applying Markov's inequality to the first of the two probability terms we get

$$\mathbb{P}\left(Y \geqslant m(1+\varepsilon)\right) = \mathbb{P}\left(e^{hY} \geqslant e^{hm(1+\varepsilon)}\right) \tag{3.19a}$$

$$\leqslant \frac{\mathbb{E}\left[e^{hY}\right]}{e^{hm(1+\varepsilon)}}. \tag{3.19b}$$

Now $\mathbb{E}\left[e^{hY}\right]$ is simply the moment generating function for $\chi^2(m)$. Using standard tables we can find that

$$\mathbb{E}\left[e^{hY}\right] = \frac{1}{(1-2h)^{m/2}} \qquad \text{for } h < \frac{1}{2}. \tag{3.20}$$

This gives us

$$\mathbb{P}\left(Y \geqslant m(1+\varepsilon)\right) \leqslant \frac{1}{(1-2h)^{m/2}} e^{-hm(1+\varepsilon)} \qquad \text{for } h < \frac{1}{2}. \tag{3.21}$$

Differentiating with respect to h to find the maximum value of the right hand side equation, we find that this occurs when

$$h = \frac{\varepsilon}{2(1+\varepsilon)} < \frac{1}{2}. \tag{3.22}$$

Substituting this value of h into Equation (3.21) we get

$$\mathbb{P}\left(Y \geqslant m(1+\varepsilon)\right) \leqslant \frac{1}{(1-2h)^{m/2}} e^{-hm(1+\varepsilon)} \qquad \text{for } h < \frac{1}{2} \tag{3.23a}$$

$$\leqslant \left((1+\varepsilon)e^{-\varepsilon}\right)^{m/2} \tag{3.23b}$$

$$\leqslant \exp\left\{-m\left(\frac{\varepsilon^2}{4} - \frac{\varepsilon^3}{6}\right)\right\}, \tag{3.23c}$$

where we go from Equation (3.23b) to Equation (3.23b) using a Taylor series expansion about

$\varepsilon = 0$,

$$((1+\varepsilon)e^{-\varepsilon})^{m/2} = 1 - \frac{m\varepsilon^2}{4} + \frac{m\varepsilon^3}{6} + \frac{1}{32}(m^2 - 4m)\varepsilon^4 + \left(\frac{m}{10} - \frac{m^2}{24}\right)\varepsilon^5 + \mathcal{O}(\varepsilon^6) \qquad (3.24a)$$

$$\leqslant \exp\left\{-m\left(\frac{\varepsilon^2}{4} - \frac{\varepsilon^3}{6}\right)\right\} \qquad (\text{since } 1 - x \leqslant e^{-x}, \ 0 \leqslant x \leqslant 1) \qquad (3.24b)$$

$$= 1 - \frac{m\varepsilon^2}{4} + \frac{m\varepsilon^3}{6} + \frac{m^2\varepsilon^4}{32} - \frac{m^2\varepsilon^5}{24} + \mathcal{O}(\varepsilon^6). \qquad (3.24c)$$

It now remains to show a similar result for the other probability term

$$\mathbb{P}\left(Y \leqslant m(1-\varepsilon)\right) = \mathbb{P}\left(e^{-hY} \geqslant e^{-hm(1-\varepsilon)}\right) \qquad (3.25a)$$

$$\leqslant \mathbb{E}\left[e^{-hY}\right] \cdot e^{hm(1-\varepsilon)} \qquad (3.25b)$$

$$= \frac{1}{(1+2h)^{m/2}} \cdot e^{hm(1-\varepsilon)}, \qquad (3.25c)$$

using the fact that

$$\mathbb{E}\left[e^{-hY}\right] = \frac{1}{(1+2h)^{m/2}}, \qquad (3.26)$$

for all $h \geqslant 0$. Again setting $h = \frac{\varepsilon}{2(1+\varepsilon)}$ gives us

$$\mathbb{P}\left(Y \leqslant m(1-\varepsilon)\right) = \left(\frac{1+\varepsilon}{1+2\varepsilon}\right)^{m/2} \exp\left\{\frac{m\varepsilon(\varepsilon-1)}{2(\varepsilon+1)}\right\}. \qquad (3.27)$$

Calculating the Taylor expansion about $\varepsilon = 0$ we find that

$$\mathbb{P}\left(Y \leqslant m(1-\varepsilon)\right) = 1 - \frac{m\varepsilon^2}{4} - \frac{m\varepsilon^3}{6} + \mathcal{O}(\varepsilon^4) \qquad (3.28a)$$

$$\leqslant 1 - \frac{m\varepsilon^2}{4} - \frac{m\varepsilon^3}{6} \qquad (3.28b)$$

$$\leqslant \exp\left\{-m\left(\frac{\varepsilon^2}{4} - \frac{\varepsilon^3}{6}\right)\right\}, \qquad (3.28c)$$

which is the same bound as for the upper tail. Putting these two tail bounds together then we get that

$$\mathbb{P}\left(\left|\|\Phi\mathbf{x}\|_2^2 - \|\mathbf{x}\|_2\right| \geqslant \varepsilon\|\mathbf{x}\|_2^2\right) = \mathbb{P}\left(\left|\frac{Y}{m} - 1\right| \geqslant \varepsilon\right) \qquad (3.29a)$$

$$= \mathbb{P}\left(Y \geqslant m(1+\varepsilon)\right) + \mathbb{P}\left(Y \leqslant m(1-\varepsilon)\right) \qquad (3.29b)$$

$$\leqslant 2\exp\left\{-m\left(\frac{\varepsilon^2}{4} - \frac{\varepsilon^3}{6}\right)\right\}. \qquad (3.29c)$$

Setting $c_0(\varepsilon) = \frac{\varepsilon^2}{4} - \frac{\varepsilon^3}{6}$ we find that this bound satisfies Equation (3.2) and hence meets the requirements of Theorem 3.2. Therefore the matrix $\Phi \in \mathbb{R}^{m \times n}$ with i.i.d. entries $\Phi_{ij} \sim \mathcal{N}\left(0, \frac{1}{m}\right)$ satisfies the restricted isometry property with probability at least $1 - e^{-cm}$, for some constant c.

3.3   Usage of the Restricted Isometry Property

In this section we will present a number of lemmas and propositions that relate to the restricted isometry property. These will be used in later chapters.

Lemma 3.3. *Suppose that $\Phi \in \mathbb{R}^{m \times n}$ obeys the restricted isometry property of order s with value δ_s. Then for any set of indices Γ such that $|\Gamma| \leqslant s$ the singular values of Φ_Γ lie in the range $[\sqrt{1 - \delta_s}, \sqrt{1 + \delta_s}]$. Furthermore the eigenvalues of $\Phi_\Gamma^\mathsf{T} \Phi_\Gamma - \mathbf{I}$ lie in the interval $[-\delta_s, \delta_s]$.*

Proof. Recall that the RIP says that for all vectors $\mathbf{x} \in \mathbb{R}^t$ for $t \leqslant s$

$$1 - \delta_s \leqslant \frac{\|\Phi_\Gamma \mathbf{x}\|_2^2}{\|\mathbf{x}\|_2^2} \leqslant 1 + \delta_s. \tag{3.30}$$

Furthermore we know that the Rayleigh quotient for a matrix A, $\rho_A(\mathbf{x})$ is bounded by the minimum and maximum eigenvalues of A, i.e. $\lambda_{min}(A) \leqslant \rho_A(\mathbf{x}) \leqslant \lambda_{max}(A)$ for all $\mathbf{x}$. Set $A = \Phi_\Gamma^\mathsf{T} \Phi_\Gamma$, then the eigenvalues of A are the square of the singular values of Φ_Γ. As the eigenvalues for A are bounded by $1 \pm \delta_s$ we have the corresponding bound for the singular values of Φ_Γ, namely $\sigma(\Phi_\Gamma) \in [\sqrt{1 - \delta_s}, \sqrt{1 + \delta_s}]$.

Let λ be an eigenvalue of A with eigenvector $\mathbf{v}$, then $A\mathbf{v} = \lambda \mathbf{v}$. Then $\lambda - 1$ is an eigenvalue of $A - \mathbf{I}$ since

$$(A - \mathbf{I})\mathbf{v} = A\mathbf{v} - \mathbf{I}\mathbf{v} = \lambda \mathbf{v} - \mathbf{v} = (\lambda - 1)\mathbf{v}. \tag{3.31}$$

Hence the eigenvalues of $\Phi_\Gamma^\mathsf{T} \Phi_\Gamma - \mathbf{I}$ lie in the interval $[-\delta_s, \delta_s]$. $\square$

Proposition 3.4 (Proposition 3.1 of (Needell and Tropp 2008)). *Suppose that $\Phi \in \mathbb{R}^{m \times n}$ has restricted isometry constant δ_s. Let Γ be a set of indices of columns of Φ such that $|\Gamma| \leqslant s$. Then for all vectors $\mathbf{y} \in \mathbb{R}^m$*

$$\left\| \Phi_\Gamma^\mathsf{T} \mathbf{y} \right\|_2 \leqslant \sqrt{1 + \delta_s} \, \|\mathbf{y}\|_2, \tag{3.32a}$$

$$\left\| \Phi_\Gamma^\dagger \mathbf{y} \right\|_2 \leqslant \frac{1}{\sqrt{1 - \delta_s}} \, \|\mathbf{y}\|_2, \tag{3.32b}$$

and $\mathbf{x} \in \mathbb{R}^s$

$$(1 - \delta_s) \, \|\mathbf{x}\|_2 \leqslant \left\| \Phi_\Gamma^\mathsf{T} \Phi_\Gamma \mathbf{x} \right\|_2 \leqslant (1 + \delta_s) \, \|\mathbf{x}\|_2, \tag{3.33a}$$

$$\frac{1}{1 - \delta_s} \leqslant \left\| \left(\Phi_\Gamma^\mathsf{T} \Phi_\Gamma \right)^{-1} \mathbf{x} \right\|_2 \leqslant \frac{1}{1 + \delta_s}. \tag{3.33b}$$

Proof. From Lemma 3.3 the singular values of Φ_Γ lie in the range $(\sqrt{1 - \delta_s}, \sqrt{1 + \delta_s})$. The bounds follow by the standard properties of singular values of matrices. $\square$

Proposition 3.5 (Approximate Orthogonality, Proposition 3.2 of (Needell and Tropp 2008)). *Suppose that Φ has restricted isometry constant δ_s and let Γ, Λ be two disjoint sets of indices such that $|\Gamma \cup \Lambda| \leqslant s$, then*

$$\left\| \Phi_\Lambda^\mathsf{T} \Phi_\Gamma \right\| \leqslant \delta_s. \tag{3.34}$$

Proof. Let $\Omega = \Lambda \cup \Gamma$. Since Λ and Γ are disjoint, $\Phi_\Lambda^\mathsf{T} \Phi_\Gamma$ is a submatrix of $\Phi_\Omega^\mathsf{T} \Phi_\Omega - \mathbf{I}$. As the singular values of a submatrix cannot exceed the maximum singular value of the full matrix we

have

$$\left\| \Phi_\Lambda^\mathsf{T} \Phi_\Gamma \right\| \leqslant \left\| \Phi_\Omega^\mathsf{T} \Phi_\Omega - \mathbf{I} \right\| \leqslant \delta_s, \tag{3.35}$$

by Lemma 3.3. $\qquad\square$

Corollary 3.6 (Corollary 3.3 of (Needell and Tropp 2008)). *Suppose that $\Phi \in \mathbb{R}^{m \times n}$ has restricted isometry constant δ_s and let Γ be a set of column indices of Φ and $\mathbf{x} \in \mathbb{R}^n$. Then if $s \geqslant |\Gamma \cup \operatorname{supp}(\mathbf{x})|$,*

$$\left\| \Phi_\Gamma^\mathsf{T} \Phi \mathbf{x}_{\Gamma^c} \right\|_2 \leqslant \delta_s \left\| \mathbf{x}_{\Gamma^c} \right\|_2. \tag{3.36}$$

Proof. Define $\Lambda = \operatorname{supp}(\mathbf{x}) \setminus \Gamma$ and as Γ, Λ are disjoint we have $\mathbf{x}_\Lambda = \mathbf{x}_{\Gamma^c}$. Then by Proposition 3.5

$$\left\| \Phi_\Gamma^\mathsf{T} \Phi \mathbf{x}_{\Gamma^c} \right\|_2 \leqslant \left\| \Phi_\Gamma^\mathsf{T} \Phi \mathbf{x}_\Lambda \right\|_2 \leqslant \left\| \Phi_\Gamma^\mathsf{T} \Phi_\Lambda \mathbf{x} \right\|_2 \leqslant \left\| \Phi_\Gamma^\mathsf{T} \Phi_\Lambda \right\|_2 \cdot \left\| \mathbf{x} \right\|_2 \leqslant \delta_s \left\| \mathbf{x} \right\|_2. \tag{3.37}$$

$\qquad\square$

Lemma 3.7 (Energy Bound, Proposition 3.5 of (Needell and Tropp 2008)). *Suppose that $\Phi \in \mathbb{R}^{m \times n}$ obeys the upper inequality of the RIP, i.e. that $\|\Phi \mathbf{v}\|_2^2 \leqslant (1 + \delta_s) \|\mathbf{v}\|_2^2$ for all s-sparse vectors $\mathbf{v}$. Then, for every signal $\mathbf{x}$,*

$$\|\Phi \mathbf{x}\|_2 \leqslant \sqrt{1 + \delta_s} \left(\|\mathbf{x}\|_2 + \frac{1}{\sqrt{s}} \|\mathbf{x}\|_1 \right). \tag{3.38}$$

Proof. We can convert the statement of this theorem into a statement about the induced norm of the operator Φ between Banach spaces. For two sets K and S we have the operator norms

$$\|\Phi\|_{K \to 2} \triangleq \max_{\mathbf{x} \in K} \|\Phi \mathbf{x}\|_2 \qquad \text{and} \qquad \|\Phi\|_{S \to 2} \triangleq \max_{\mathbf{x} \in S} \|\Phi \mathbf{x}\|_2. \tag{3.39}$$

Let $I \subset \{1, 2, \dots, n\}$ then write $\mathcal{B}_{\ell_2}^I$ for the ℓ_2 unit ball restricted to the components of $\mathbb{R}^n$ in I. Define the convex body

$$S \triangleq \operatorname{conv} \left(\bigcup_{|I| \leqslant s} \mathcal{B}_{\ell_2}^I \right), \tag{3.40}$$

which is the convex hull of all s-sparse ℓ_2 balls in $\mathbb{R}^n$.

Any $\mathbf{v} \in S$ can be written in the form $\mathbf{v} = \sum_{i=1} \lambda_i \mathbf{x}_i$ such that $\sum_i |\lambda_i| \leqslant 1$ where x_i is the unique vector in the i-th ball and $\|x\|_2 = 1$, that is, we write $\mathbf{v}$ as the convex sum of s-sparse unit vectors. Then

$$\|\Phi \mathbf{v}\|_2^2 = \left\| \Phi \left(\sum_i \lambda_i \mathbf{x}_i \right) \right\|_2^2 \tag{3.41a}$$

$$\leqslant \sum_i |\lambda_i| \, \|\Phi \mathbf{x}_i\|_2^2 \tag{3.41b}$$

$$\leqslant \sum_i |\lambda_i| \, (1 + \delta_s) \, \|\mathbf{x}_i\|_2^2 \tag{3.41c}$$

$$\leqslant \sum_i |\lambda_i| \, (1 + \delta_s) \tag{3.41d}$$

$$\leqslant 1 + \delta_s. \tag{3.41e}$$

Hence by the hypothesis we have $\|\Phi\|_{S \to 2} \leqslant \sqrt{1 + \delta_s}$. Define a second convex body

$$K \triangleq \left\{ \mathbf{x} : \|\mathbf{x}\|_2 + \frac{1}{\sqrt{s}} \|\mathbf{x}\|_1 \leqslant 1 \right\}. \tag{3.42}$$

The statement of the lemma then becomes

$$\|\Phi\|_{K\to 2} \leqslant \|\Phi\|_{S\to 2}, \tag{3.43}$$

and it suffices to check the condition that $K \subset S$. To this end, assume that $\mathbf{x} \in K$. Partition the support of $\mathbf{x}$ into sets I_j of size s such that I_0 is the set of the largest s components of $\mathbf{x}$, I_1 is the set of the next largest s components, and so on. Any ties should be broken deterministically, e.g. lexicographically. Let I_J be the final block which may have fewer than s entries. Then we can decompose $\mathbf{x}$ over these sets I_i

$$\mathbf{x} = \mathbf{x}_{I_0} + \sum_{j=1}^{J} \mathbf{x}_{I_j} = \lambda_0 \mathbf{y}_0 + \sum_{j=1}^{J} \lambda_j \mathbf{y}_j, \tag{3.44}$$

where

$$\lambda_j = \left\|\mathbf{x}_{I_j}\right\|_2 \quad \text{and} \quad \mathbf{y}_j = \frac{\mathbf{x}_{I_j}}{\left\|\mathbf{x}_{I_j}\right\|_2}. \tag{3.45}$$

Observe that each $\mathbf{y}_j$ is s-sparse and has unit ℓ_2 norm. We will show that $\mathbf{x}$ can be written as a convex combination of vectors from S, namely that $\sum_{j=0}^{J} \lambda_j \leqslant 1$.

Now fix $j \in \mathbb{Z}$ where $1 \leqslant j \leqslant J$, then I_j has at most s elements and I_{j-1} has exactly s elements. Then $\left\|\mathbf{x}_{I_j}\right\|_2 \leqslant \sqrt{s} \cdot \left\|\mathbf{x}_{I_j}\right\|_\infty$ and $\left\|\mathbf{x}_{I_{j-1}}\right\|_1 \geqslant r \cdot \left\|\mathbf{x}_{I_j}\right\|_\infty$ since each of the r components of $\mathbf{x}_{I_{j-1}}$ is as least as large as the largest component of $\mathbf{x}_{I_j}$. Thus

$$\lambda_j = \left\|\mathbf{x}_{I_j}\right\|_2 \leqslant \sqrt{s} \left\|\mathbf{x}_{I_j}\right\|_\infty \leqslant \frac{1}{\sqrt{s}} \left\|\mathbf{x}_{I_{j-1}}\right\|_1. \tag{3.46}$$

Summing these terms over j we get

$$\sum_{j=1}^{J} \lambda_j \leqslant \frac{1}{\sqrt{s}} \sum_{j=1}^{J} \left\|\mathbf{x}_{I_{j-1}}\right\|_1 = \frac{1}{\sqrt{s}} \left\|\mathbf{x}_{\bigcup_{j=0}^{J-1} I_j}\right\|_1 \leqslant \frac{1}{\sqrt{s}} \|\mathbf{x}\|_1. \tag{3.47}$$

Since $\lambda_0 = \left\|\mathbf{x}_{I_0}\right\|_2 \leqslant \|\mathbf{x}\|_2$ and $\mathbf{x} \in K$ implies that $\|\mathbf{x}\|_2 + \frac{1}{\sqrt{s}} \|\mathbf{x}\|_1 \leqslant 1$ we have

$$\sum_{j=0}^{J} \lambda_j \leqslant \|\mathbf{x}\|_2 + \frac{1}{\sqrt{s}} \|\mathbf{x}\|_1 \leqslant 1, \tag{3.48}$$

so we have written $\mathbf{x}$ as a convex combination of s-sparse unit vectors, hence $\mathbf{x} \in S$. $\square$

Lemma 3.8 (Reduction to sparse case, Lemma 6.1 of (Needell and Tropp 2008)). *Let $\mathbf{x}$ be a vector from $\mathbb{R}^n$ and assume that Φ obeys the RIP of order s with $\delta_s \leqslant 0.1$, then the sample vector $\mathbf{y} = \Phi\mathbf{x} + \mathbf{e}$ can also be written as $\mathbf{y} = \Phi\mathbf{x}^s + \tilde{\mathbf{e}}$ where*

$$\|\tilde{\mathbf{e}}\|_2 \leqslant 1.05 \left(\|\mathbf{x} - \mathbf{x}^s\|_2 + \frac{1}{\sqrt{s}} \|\mathbf{x} - \mathbf{x}^s\|_1 \right) + \|\mathbf{e}\|_2. \tag{3.49}$$

Proof. Writing $\mathbf{x} = \mathbf{x}^s + (\mathbf{x} - \mathbf{x}^s)$ we get $\mathbf{y} = \Phi\mathbf{x}^s + \tilde{\mathbf{e}}$ where $\tilde{\mathbf{e}} = \Phi(\mathbf{x} - \mathbf{x}^s) + \mathbf{e}$. Then from Lemma 3.7 we get

$$\|\tilde{\mathbf{e}}\|_2 \leqslant \sqrt{1 + \delta_s} \left(\|\mathbf{x} - \mathbf{x}^s\|_2 + \frac{1}{\sqrt{s}} \|\mathbf{x} - \mathbf{x}^s\|_1 \right) + \|\mathbf{e}\|_2. \tag{3.50}$$

Then we use the fact that $\delta_s \leqslant 0.1$ to get $\sqrt{1 + \delta_s} \leqslant 1.05$. $\square$

For the following two lemmas, Lemmas 3.10 and 3.11 we need the modified restricted isometry property , which is used in section 5.2.1.

Definition 3.9. *Let $\hat{\Phi}$ be a matrix satisfying the restricted isometry property for δ_s and let $\Phi = \frac{\hat{\Phi}}{1+\delta_s}$, then*

$$(1 - \beta_s) \|\mathbf{x}\|_2^2 \leqslant \|\Phi\mathbf{x}\|_2^2 \leqslant \|\mathbf{x}\|_2, \tag{3.51}$$

for all s-sparse $\mathbf{x}$, where

$$\beta_s = 1 - \frac{1 - \delta_s}{1 + \delta_s} < 1, \tag{3.52}$$

and we say that Φ obeys the modified restricted isometry property *of order s.*

Lemma 3.10. *For all index sets Γ and all measurement matrices Φ for which the RIP holds with $s = |\Gamma|$*

$$\left\| \left(\mathbf{I} - \Phi_\Gamma^\mathsf{T} \Phi_\Gamma \right) \mathbf{x}_\Gamma \right\|_2 \leqslant \delta_s \|\mathbf{x}_\Gamma\|_2. \tag{3.53}$$

and furthermore

$$\left\| \left(\mathbf{I} - \gamma\Phi_\Gamma^\mathsf{T} \Phi_\Gamma \right) \mathbf{x}_\Gamma \right\|_2 \leqslant [1 - \gamma(1 - \delta_s)] \|\mathbf{x}_\Gamma\|_2. \tag{3.54}$$

Or if Φ obeys the modified RIP we have

$$\left\| \left(\mathbf{I} - \Phi_\Gamma^\mathsf{T} \Phi_\Gamma \right) \mathbf{x}_\Gamma \right\|_2 \leqslant \beta_s \|\mathbf{x}_\Gamma\|_2. \tag{3.55}$$

and

$$\left\| \left(\mathbf{I} - \gamma\Phi_\Gamma^\mathsf{T} \Phi_\Gamma \right) \mathbf{x}_\Gamma \right\|_2 \leqslant [1 - \gamma(1 - \beta_s)] \|\mathbf{x}_\Gamma\|_2. \tag{3.56}$$

Proof. The RIP guarantees us that the eigenvalues of $\Phi_\Gamma^\mathsf{T}\Phi_\Gamma$ lie in the range $1 - \delta_s$ to $1 + \delta_s$. Hence the matrix $\mathbf{I} - \Phi_\Gamma^\mathsf{T}\Phi_\Gamma$ has eigenvalues in $[-\delta_s, \delta_s]$. For the second result we proceed similarly. Clearly $\gamma\Phi_\Gamma^\mathsf{T}\Phi_\Gamma$ has eigenvalues in the range $[\gamma(1 - \delta_s), \gamma(1 + \delta_s)]$, hence the maximum eigenvalue of $\mathbf{I} - \gamma\Phi_\Gamma^\mathsf{T}\Phi_\Gamma$ is less than or equal to $1 - \gamma(1 - \delta_s)$. Then for each fixed Γ we have

$$\left\| \left(\mathbf{I} - \gamma\Phi_\Gamma^\mathsf{T} \Phi_\Gamma \right) \mathbf{x}_\Gamma \right\|_2 \leqslant \left\| \left(\mathbf{I} - \gamma\Phi_\Gamma^\mathsf{T} \Phi_\Gamma \right) \right\|_2 \cdot \|\mathbf{x}_\Gamma\|_2 \tag{3.57a}$$

$$\leqslant \max_{\Lambda: |\Lambda|=s} \left\| \left(\mathbf{I} - \gamma\Phi_\Lambda^\mathsf{T} \Phi_\Lambda \right) \right\|_2 \cdot \|\mathbf{x}_\Gamma\|_2 \tag{3.57b}$$

$$\leqslant (1 - \gamma(1 - \delta_s)) \|\mathbf{x}_\Gamma\|_2, \tag{3.57c}$$

and Equation (3.53) follows by setting $\gamma = 1$. The results for the modified RIP follow analogously, completing the proof. $\square$

Lemma 3.11 (Lemma 2 of (Blumensath and Davies 2008)). *Let Γ and Λ be two disjoint index sets for the matrix Φ. Then for all Φ for which the RIP holds with $s = |\Gamma \cup \Lambda|$*

$$\left\| \Phi_\Gamma^\mathsf{T} \Phi_\Lambda \mathbf{x}_\Lambda \right\|_2 \leqslant \delta_s \|\mathbf{x}_\Lambda\|_2. \tag{3.58}$$

Or if Φ obeys the modified RIP we have

$$\left\| \Phi_\Gamma^\mathsf{T} \Phi_\Lambda \mathbf{x}_\Lambda \right\|_2 \leqslant \beta_s \|\mathbf{x}_\Lambda\|_2. \tag{3.59}$$

Proof. Set $\Omega = \Gamma \cup \Lambda$. Since Γ and Λ are disjoint, the matrix $\left(-\Phi_\Gamma^\mathsf{T} \Phi_\Lambda\right)$ is a submatrix of the matrix $\mathbf{I} - \Phi_\Omega^\mathsf{T} \Phi_\Omega$. Since the largest singular value of a submatrix is bounded above by the largest singular value of the full matrix, we have

$$\left\| \Phi_\Gamma^\mathsf{T} \Phi_\Lambda \right\|_2 \leqslant \left\| \mathbf{I} - \Phi_\Omega^\mathsf{T} \Phi_\Omega \right\|_2 \leqslant \delta_s \tag{3.60}$$

since by Lemma 3.3 we know that the eigenvalues of $\mathbf{I} - \Phi_\Omega^\mathsf{T} \Phi_\Omega$ are bounded above by δ_s. The statement for the modified RIP follows similarly, completing the lemma. $\qquad\square$

Although not making use of the restricted isometry property, we nonetheless include the following as we require it later in the proof relating to the CoSaMP algorithm. This lemma says that the ℓ_2 norm of the tail of a signal is much smaller than the ℓ_1 norm of the entire signal.

Lemma 3.12 (Lemma 7 of (Gilbert, Strauss, Tropp, and Vershynin 2007)). *Let $\mathbf{x} \in \mathbb{R}^n$ be a signal and denote the best t-term approximation by $\mathbf{x}^t$. Then it holds that*

$$\left\| \mathbf{x} - \mathbf{x}^t \right\|_2 \leqslant \frac{1}{2\sqrt{t}} \left\| \mathbf{x} \right\|_1 . \tag{3.61}$$

Proof. Observe that for any vector $\mathbf{z} \in \mathbb{R}^n$ we have $\|\mathbf{z}\|_2 \leqslant \sqrt{\|\mathbf{z}\|_1 \cdot \|\mathbf{z}\|_\infty}$ and $\|\mathbf{z}\|_1 = \left\|\mathbf{z}^t\right\|_1 + \left\|\mathbf{z} - \mathbf{z}^t\right\|_1$. Now let μ denote the magnitude of the $(t+1)$-th largest component of $\mathbf{x}$. If $\mu = 0$ then $\left\|\mathbf{x} - \mathbf{x}^t\right\|_2 = 0$ and we are done. Hence assume that $\mu > 0$ and we perform the following calculations

$$\frac{\left\|\mathbf{x} - \mathbf{x}^t\right\|_2}{\|\mathbf{x}\|_1} \leqslant \frac{\sqrt{\left\|\mathbf{x} - \mathbf{x}^t\right\|_1 \cdot \left\|\mathbf{x} - \mathbf{x}^t\right\|_\infty}}{\left\|\mathbf{x}^t\right\|_1 + \left\|\mathbf{x} - \mathbf{x}^t\right\|_1} \tag{3.62a}$$

$$\leqslant \frac{\sqrt{\mu \left\|\mathbf{x} - \mathbf{x}^t\right\|_1}}{t\mu + \left\|\mathbf{x} - \mathbf{x}^t\right\|_1}. \tag{3.62b}$$

Define the function $f(x) \triangleq \frac{\sqrt{bx}}{c+x}$ which has maximum at $x = c$, namely $f(c) = \frac{\sqrt{b}}{2\sqrt{c}}$. Set $c = t\mu$ and $b = \mu$ so the above equation becomes $f(x)$ where $x = \left\|\mathbf{x} - \mathbf{x}^t\right\|_1$. Thus we get

$$\frac{\left\|\mathbf{x} - \mathbf{x}^t\right\|_2}{\|\mathbf{x}\|_1} \leqslant \frac{\sqrt{\mu \left\|\mathbf{x} - \mathbf{x}^t\right\|_1}}{t\mu + \left\|\mathbf{x} - \mathbf{x}^t\right\|_1} \tag{3.63a}$$

$$\leqslant \frac{\sqrt{t\mu^2}}{2\sqrt{t\mu}} \tag{3.63b}$$

$$= \frac{1}{2\sqrt{t}}, \tag{3.63c}$$

completing the proof of the lemma. $\qquad\square$

3.4 Alternatives to the Restricted Isometry Property

There are a number of properties similar to the restricted isometry property for matrices which are used in some papers. Here we will present a brief overview of some of these other properties of matrices.

3.4.1 Uniform Uncertainty Principle

The Uniform Uncertainty Principle or UUP is a statement about the minimum and maximum eigenvalues of the matrix $\Phi^\mathsf{T}\Phi$ and is similar, in principle, to results such as the Johnson-Lindenstrauss lemma (Johnson and Lindenstrauss 1984) regarding the preservation of distances between a finite number of points randomly projected onto a lower dimensional space.

Definition 3.13 (Uniform Uncertainty Principle, (Candès and Tao 2006)). *For a measurement matrix $\Phi \in \mathbb{R}^{m \times n}$, we say that Φ obeys the* uniform uncertainty principle *or UUP with oversampling factor λ if for every sufficiently small $\alpha > 0$, the following statement is true with probability at least $1 - \mathcal{O}(n^{-\rho/\alpha})$ for some fixed positive constant $\rho > 0$; for all subsets Γ such that*

$$|\Gamma| \leqslant \alpha \cdot \frac{m}{\lambda}, \tag{3.64}$$

the matrix Φ_Γ obeys the bounds

$$\frac{m}{2n} \leqslant \lambda_{\min}\left(\Phi_\Gamma^\mathsf{T}\Phi_\Gamma\right) \leqslant \lambda_{\max}\left(\Phi_\Gamma^\mathsf{T}\Phi_\Gamma\right) \leqslant \frac{3m}{2n}. \tag{3.65}$$

Note that Equation (3.65) is equivalent to

$$\frac{m}{2n} \leqslant \frac{\|\Phi_\Gamma \mathbf{x}\|_2^2}{\|\mathbf{x}\|_2^2} \leqslant \frac{3m}{2n}, \tag{3.66}$$

holding for all signals $\mathbf{x}$ with support size less than or equal to $\alpha m/\lambda$. This in turn is is equivalent to the RIP setting

$$\delta = \max\left\{\frac{m}{2n}, \frac{3m}{2n} - 1\right\}. \tag{3.67}$$

The authors of (Candès and Tao 2006) make the point that in fact the constants in Equation (3.65) can be something other than $1/2$ and $3/2$. In fact any two values a and b would suffice provided that $0 < a, b < \infty$. The values of $1/2$ and $3/2$ are selected only for concreteness. The name "uniform uncertainty principle" comes from the case when Φ is a Fourier matrix. Consider a sparse signal f such that $|\mathrm{supp}(f)| \leqslant \alpha m/\lambda$ with discrete Fourier transform $\hat{f}$, from Parseval's theorem we have that $\left\|\hat{f}\right\|_2 = \|f\|_2$. From Equation (3.65) we then get

$$\left\|\hat{f}\right\|_2 \leqslant \sqrt{\frac{3m}{2n}} \|f\|_2, \tag{3.68}$$

which says that f cannot be concentrated in frequency unless m is comparable to n.

3.4.2 Exact Reconstruction Principle

Definition 3.14 (Exact Reconstruction Principle, (Candès and Tao 2006)). *We say that a measurement matrix $\Phi \in \mathbb{R}^{m \times n}$ obeys the* exact reconstruction principle *(ERP) with oversampling factor λ if for all sufficiently small $\alpha > 0$, each fixed subset Γ obeying*

$$|\Gamma| \leqslant \alpha \cdot \frac{m}{\lambda}, \tag{3.69}$$

and each sign sequence σ defined on Γ, there exists with overwhelmingly large probability a vector $\mathbf{x} \in \mathbb{R}^n$ with the following properties:

(a) $\mathbf{x}_\Gamma$ *has the same sign sequence as σ;*

(b) $\mathbf{x}$ *is a linear combination of the rows of* Φ*, i.e.* $\mathbf{x} = \Phi \mathbf{z}$ *for some* $\mathbf{z}$*;*

(c) $\|\mathbf{x}_{\Gamma^c}\|_\infty \leqslant \frac{1}{2}$*.*

By "overwhelmingly large" we mean that the probability be at least $1 - \mathcal{O}\left(n^{-\rho/\alpha}\right)$ for some fixed positive constant $\rho > 0$. The ERP is primarily used to check that the vector obtained by ℓ_1 minimisation is close to the sparse approximation to the original signal in the ℓ_1-norm, for more details see (Candès and Tao 2006).

These two definitions are used primarily in the hypothesis of Theorem 2.13. It is shown in (Candès and Tao 2006) that matrices drawn from the Fourier ensemble or with i.i.d. Gaussian or binary random entries satisfy the ERP and UUP. It can also be shown that under some conditions, the UUP implies the ERP.

4

Frames

Frames are a generalisation of a basis for Hilbert spaces. For a Hilbert space $\mathcal{H}$ a frame may be loosely regarded as a set of elements $\{x_i\}$, $x_i \in \mathcal{H}$ where the x_i span the space $\mathcal{H}$. Since we used frames as the inspiration for our reconstruction algorithm we will present a background to frames in this chapter.

4.1 Definition

Definition 4.1. *A sequence $\{x_n\}$ in a Hilbert space $\mathcal{H}$ is a* frame *if there exist numbers $A, B > 0$ such that for all $x \in \mathcal{H}$ we have*

$$A \left\| x \right\|^2 \leqslant \sum_n |\langle x, x_n \rangle|^2 \leqslant B \left\| x \right\|^2. \tag{4.1}$$

The numbers A and B are called the frame bounds *and the frame is said to be* tight *if $A = B$. The frame is* exact *if it ceases to be a frame whenever any single element is deleted from the sequence.*

It follows that every orthonormal sequence is a tight exact frame with $A = B = 1$. Furthermore as the summand above is a series of positive real numbers, if it converges, it converges absolutely and hence unconditionally. That is, any rearrangement of the terms will also converge and to the same value.

Let $\{x_n\}$ be a frame for a Hilbert space $\mathcal{H}$. Then any $x \in \mathcal{H}$ can be written as

$$x = \sum c_n x_n. \tag{4.2}$$

Unlike for a basis, this representation is not necessarily unique, as the x_n are not required to be linearly independent. We give an example of a frame for $\mathbb{R}^2$ in Example 4.2.

Example 4.2. *The set of vectors*

$$\mathcal{F} = \left\{ \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} \sqrt{3} \\ -1 \end{pmatrix}, \begin{pmatrix} -\sqrt{3} \\ -1 \end{pmatrix} \right\}, \tag{4.3}$$

span $\mathbb{R}^2$ *and hence form a frame for* $\mathbb{R}^2$. *Forming the matrix* $\Phi_{\mathcal{F}}$ *with the elements of* $\mathcal{F}$ *as columns gives*

$$\Phi_{\mathcal{F}} = \begin{pmatrix} 0 & \sqrt{3} & -\sqrt{3} \\ 1 & -1 & -1 \end{pmatrix}, \tag{4.4}$$

and maximum and minimum eigenvalues of $\Phi_{\mathcal{F}}\Phi_{\mathcal{F}}^*$, *give the frame bounds,* $A = 3$ *and* $B = 6$.

We make the following definitions about an operators between Hilbert spaces. (Heil and Walnut 1989)

Definition 4.3. *Let* $\mathcal{H}$ *and* $\mathcal{G}$ *be Hilbert spaces with norms* $\|\cdot\|_{\mathcal{H}}$ *and* $\|\cdot\|_{\mathcal{G}}$ *and inner products* $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ *and* $\langle \cdot, \cdot \rangle_{\mathcal{G}}$ *defined respectively. Let* $S : \mathcal{H} \longrightarrow \mathcal{G}$.

(a) *S is* linear *if* $S(ax + by) = aS(x) + bS(y)$ *for all* $x, y \in \mathcal{H}$ *and* $a, b \in \mathbb{C}$.

(b) *S is* one-to-one *or* injective *if* $Sx \neq Sy$ *whenever* $x \neq y$ *and* $x, y \in \mathcal{H}$.

(c) *The* range *or* image *of S is* $\mathfrak{Im}(S) \triangleq \{ Sx \mid x \in \mathcal{H} \}$.

(d) *S is* onto *or* surjective *if* $\mathfrak{Im}(S) = \mathcal{G}$.

(e) *The* norm *of S is* $\|S\| \triangleq \sup \{ \|Sx\|_{\mathcal{G}} \mid x \in \mathcal{H}, \|x\|_{\mathcal{H}} = 1 \}$.

(f) *S is* continuous *if* $x_n \longrightarrow x$ *implies* $Sx_n \longrightarrow Sx$.

(g) *S is* bounded *if* $\|S\| < \infty$. *A linear operator is bounded if and only if it is continuous.*

(h) *The* adjoint *of S is the unique operator* $S^* : \mathcal{G} \longrightarrow \mathcal{H}$ *such that* $\langle Sx, y \rangle_{\mathcal{G}} = \langle x, S^*y \rangle_{\mathcal{H}}$ *for all* $x \in \mathcal{H}$ *and* $y \in \mathcal{G}$. *Furthermore* $\|S^*\| = \|S\|$.

(i) *If S is* bijective *then it has inverse* $S^{-1} : \mathcal{G} \longrightarrow \mathcal{H}$ *defined by* $S^{-1}y \triangleq x$ *if* $y = Sx$.

(j) *S is* invertible *if S is linear, bijective, continuous and* S^{-1} *is continuous. It follows then that*

$$\left\| S^{-1} \right\|^{-1} \|x\|_{\mathcal{H}} \leqslant \|Sx\|_{\mathcal{G}} \leqslant \|S\| \, \|x\|_{\mathcal{H}}$$

for all $x \in \mathcal{H}$.

(k) *S is an* isometry *or* norm-preserving *if* $\|Sx\|_{\mathcal{G}} = \|x\|_{\mathcal{H}}$ *for all* $x \in \mathcal{H}$. *If S is linear then it is an isometry if and only if* $\langle Sx, Sy \rangle_{\mathcal{G}} = \langle x, y \rangle_{\mathcal{H}}$ *for all* $x, y \in \mathcal{H}$.

(l) *S is* unitary *if it is a linear bijective isometry.*

In the special case where $\mathcal{G} = \mathcal{H}$, that is $S : \mathcal{H} \longrightarrow \mathcal{H}$ we have the following.

(a) *S is* self-adjoint *if* $S = S^*$, *that is,* $\langle Sx, y \rangle_{\mathcal{H}} = \langle x, Sy \rangle_{\mathcal{H}}$ *for all* $x, y \in \mathcal{H}$.

(b) *S is* positive, *written* $S \geqslant 0$, *if* $\langle Sx, x \rangle_{\mathcal{H}} \geqslant 0$ *for all* $x \in \mathcal{H}$. *If S is positive then it is also self-adjoint.*

4.2 Properties

We will now demonstrate several properties of frames, in particular, relating to the frame bounds. Let us define the function $S : \mathcal{H} \to \mathcal{H}$, called the *frame operator*, by

$$Sf \triangleq \sum_j \langle f, x_j \rangle x_j,$$

where the $\{x_j\}$ form a frame for $\mathcal{H}$. We are now in a position to state the following theorem.

Theorem 4.4. *Given a sequence $\{x_n\}$ in a Hilbert space $\mathcal{H}$, the following two statements are equivalent:*

(a) $\{x_n\}$ *is a frame with bounds A, B.*

(b) *The frame operator $Sx = \sum \langle x, x_n \rangle x_n$ is a bounded linear operator with*

$$AI \leqslant S \leqslant BI,$$

where I is the identity operator.

Proof. We first show that (b) implies (a). If (b) holds, then for all $x \in \mathcal{H}$

$$\langle AIx, x \rangle = A \langle x, x \rangle \leqslant \langle Sx, x \rangle \leqslant \langle BIx, x \rangle = B \langle x, x \rangle . \tag{4.5}$$

But $\langle Sx, x \rangle = \sum |\langle x, x_n \rangle|$ and $\langle x, x \rangle = \|x\|^2$, hence (a) is true.

We now show that (a) implies (b). Fix an element $x \in \mathcal{H}$ and let s_N be the partial frame operator for x, that is

$$s_N = \sum_{n=-N}^{N} \langle x, x_n \rangle x_n. \tag{4.6}$$

Then for $M \leqslant N$ and by the Cauchy-Schwarz inequality for series, we have that

$$\|s_N - s_M\|^2 = \sup_{\|y\|=1} |\langle s_N - s_M, y \rangle|^2 \tag{4.7a}$$

$$= \sup_{\|y\|=1} \left| \sum_{M<|n|<N} \langle x, x_n \rangle \langle x_n, y \rangle \right|^2 \tag{4.7b}$$

$$\leqslant \sup_{\|y\|=1} \left(\sum_{M<|n|<N} |\langle x, x_n \rangle|^2 \right) \left(\sum_{M<|n|<N} |\langle x_n, y \rangle|^2 \right) \tag{4.7c}$$

$$\leqslant \sup_{\|y\|=1} \left(\sum_{M<|n|<N} |\langle x, x_n \rangle|^2 \right) B \|y\|^2 \tag{4.7d}$$

$$= B \sum_{M<|n|<N} |\langle x, x_n \rangle|^2 \tag{4.7e}$$

$$\longrightarrow 0 \quad \text{as } M, N \longrightarrow \infty. \tag{4.7f}$$

Hence the sequence $\{s_N\}$ forms a Cauchy sequence in $\mathcal{H}$ and thus must converge. Therefore Sx is a well-defined element of $\mathcal{H}$ as it is the limit of this sequence. Repeating the process above

we show that the operator S is bounded

$$\|Sx\|^2 = \sup_{\|y\|=1} |\langle Sx, y\rangle|^2 \tag{4.8a}$$

$$= \sup_{\|y\|=1} \left| \left\langle \sum_n \langle x, x_n\rangle x_n, y \right\rangle \right|^2 \tag{4.8b}$$

$$\leqslant \sup_{\|y\|=1} \left| \sum_n \langle x, x_n\rangle \langle x_n, y\rangle \right|^2 \tag{4.8c}$$

$$\leqslant \sup_{\|y\|=1} \left(\sum_n |\langle x, x_n\rangle|^2 \right) \left(\sum_n |\langle x_n, y\rangle|^2 \right) \tag{4.8d}$$

$$\leqslant \sup_{\|y\|=1} \|x\|^2 \cdot B \cdot \|y\|^2 \tag{4.8e}$$

$$= B \|x\|^2, \tag{4.8f}$$

hence S is bounded by B. Then (b) follows from the definition of the frame bounds. $\qquad\square$

Corollary 4.5. *Under the hypothesis of Theorem 4.4 we have:*

(a) *S is invertible and $B^{-1}I \leqslant S^{-1} \leqslant A^{-1}I$.*

(b) *$\{S^{-1}x_n\}$ is a frame with frame bounds $B^{-1} \leqslant A^{-1}$ and is called the* dual frame *of $\{x_n\}$.*

(c) *Every $x \in \mathscr{H}$ can be written*

$$x = \sum \left\langle x, S^{-1}x_n \right\rangle x_n = \sum \langle x, x_n\rangle S^{-1}x_n. \tag{4.9}$$

4.3   Frame Reconstruction Algorithm

Assume that $\{x_n\}_n$ forms a frame for the Hilbert space $\mathscr{H}$ with frame bounds $0 < A \leqslant B < \infty$. Then every $f \in \mathscr{H}$ is uniquely determined by the sequence $\{\langle f, x_n\rangle\}_n$. Then the following, Algorithm 4.1, is a reconstruction algorithm for $f \in \mathscr{H}$ given the sequence $\{\langle f, x_n\rangle\}_n$. Unfortunately we cannot just project back onto the frame elements as we can for a basis, i.e.

$$f \neq \sum_n x_n \langle f, x_n\rangle, \tag{4.10}$$

in general, as we show in Example 4.6.

Example 4.6. *Consider the Hilbert space $\mathscr{H} = \mathbb{R}^2$ with frame elements $\{\mathbf{e}_1, \mathbf{e}_1, \mathbf{e}_2\}$ where $\mathbf{e}_i$ is the canonical basis element, that is $\mathbf{e}_i$ is a vector of zeroes except in the i-th component where it is equal to 1. Then any vector $f \in \mathscr{H}$ with components x and y gives the sequence of inner products $\{x, x, y\}$. If we reconstruct using Equation (4.10) we get the vector $(2x\ y)^\mathsf{T} \neq f$.*

Algorithm 4.1 Frame Reconstruction Algorithm

Input: The sequence of inner products $\{\langle f, \mathbf{x}_n \rangle\}_j$.

Output: The function $f \in \mathscr{H}$

1: $f^{(0)} \leftarrow 0$
2: **for** $j = 1, \ldots$ **do**
3:    $f^{(j)} \leftarrow f^{(j-1)} + \frac{2}{A+B} S \left(f - f^{(j-1)} \right)$
4: **end for**
5: **return** $f^{(\infty)}$

Lemma 4.7. *Let $\{x_n\}$ be a frame for the Hilbert space $\mathscr{H}$ with frame bounds $0 < A \leqslant B < \infty$. Then Algorithm 4.1 reconstructs every $f \in \mathscr{H}$ from the inner products $\{\langle f, x_n \rangle\}$. Furthermore we have the error estimate*

$$\left\| f - f^{(j)} \right\|_2 \leqslant \left(\frac{B-A}{B+A} \right)^j \|f\|_2 . \tag{4.11}$$

Proof. Firstly observe that for all $f \in \mathscr{H}$

$$\langle Sf, f \rangle = \left\langle \sum_n \langle f, x_n \rangle x_n, f \right\rangle \tag{4.12a}$$

$$= \sum_n \langle \langle f, x_n \rangle x_n, f \rangle \tag{4.12b}$$

$$= \sum_n \langle f, x_n \rangle \langle f, x_n \rangle \tag{4.12c}$$

$$= \sum_n |\langle f, x_n \rangle|^2 . \tag{4.12d}$$

Therefore by the frame bounds we have

$$A \|f\|_2 \leqslant \sum_n |\langle f, x_n \rangle|^2 \leqslant B \|f\|_2 , \tag{4.13}$$

for all $f \in \mathscr{H}$. Consider the operator $\left(\mathbf{1} - \frac{2}{A+B} S \right)$, then

$$\left\langle \left(\mathbf{1} - \frac{2}{A+B} S \right) f, f \right\rangle = \langle f, f \rangle - \frac{2}{A+B} \langle Sf, f \rangle \tag{4.14a}$$

$$\geqslant \|f\|_2^2 - \frac{2}{A+B} B \|f\|_2^2 \tag{4.14b}$$

$$= \frac{A-B}{A+B} \|f\|_2^2 . \tag{4.14c}$$

Similarly we calculate an upper bound to get

$$-\frac{B-A}{B+A} \|f\|_2^2 \leqslant \left\langle \left(\mathbf{1} - \frac{2}{A+B} S \right) f, f \right\rangle \leqslant \frac{B-A}{B+A} \|f\|_2^2 . \tag{4.15}$$

Therefore we have the operator norm

$$\left\| \mathbf{1} - \frac{2}{A+B} S \right\| \leqslant \frac{B-A}{B+A} < 1 . \tag{4.16}$$

From the update step of the algorithm we have

$$f - f^{(n)} = f - f^{(n-1)} - \frac{2}{A+B} S \left(f - f^{(n-1)} \right) \tag{4.17a}$$

$$= \left(\mathbf{1} - \frac{2}{A+B} S \right) \left(f - f^{(n-1)} \right), \tag{4.17b}$$

and hence

$$f - f^{(n)} = \left(\mathbf{1} - \frac{2}{A+B} S \right)^n \left(f - f^{(0)} \right). \tag{4.18}$$

But by the bound on the operator we have

$$\left\| f - f^{(n)} \right\|_2 \leqslant \left\| \mathbf{1} - \frac{2}{A+B} S \right\|^n \left\| f - f^{(0)} \right\|_2 \tag{4.19a}$$

$$\leqslant \left(\frac{B-A}{B+A} \right)^n \| f \|_2, \tag{4.19b}$$

as $f^{(0)} \equiv 0$. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\square$

Furthermore this algorithm will converge to f for *any* starting vector $f^{(0)}$. Algorithms that converge faster are known to exist as in (Gröchenig 1993), but all follow this basic idea.

Assume that we are dealing with the Hilbert space $\mathscr{H} = \mathbb{R}^n$ for some n and let $\{ \mathbf{x}_j \}$ be a frame for $\mathscr{H}$. Necessarily there are $s \geqslant n$ of the $\mathbf{x}_j$. Let $\Theta \in \mathbb{R}^{s \times n}$ be the matrix formed by taking the $\mathbf{x}_j$ as rows. Then we can rewrite the frame operator for a vector $\mathbf{y} \in \mathbb{R}^n$ in matrix notation as

$$S\mathbf{y} = \sum_{j=1}^{s} \langle \mathbf{y}, \mathbf{x}_j \rangle \, \mathbf{x}_j = \Theta^\mathsf{T} \Theta \mathbf{y}. \tag{4.20}$$

Then the update step of the frame reconstruction algorithm becomes

$$\mathbf{y}^{(i)} \leftarrow \mathbf{y}^{(i-1)} + \frac{2}{A+B} \Theta^\mathsf{T} \Theta \left(\mathbf{y} - \mathbf{y}^{(i-1)} \right). \tag{4.21}$$

Since $\mathbf{y}$ is not known, but the measurements $\left\{ \langle \mathbf{y}, \mathbf{x}_j \rangle \right\}_j = \Theta \mathbf{y}$ are, this can in fact be calculated.

Part III

Reconstruction Algorithms

Reconstruction Algorithms

Originally compressive sensing relied on the assumption that the solution to the ℓ_1 minimisation problem provides the correct solution and is computationally feasible. However work has been done to find alternative algorithms that are faster or give superior reconstruction performance.

In this chapter we will present an overview of existing algorithms to reconstruct the signal $\mathbf{x}$ from the measured signal $\mathbf{y} = \Phi\mathbf{x}(+\mathbf{e})$ (for some error vector $\mathbf{e}$). Many of the algorithms we present fall in two main categories: variations on matching pursuit techniques and thresholding algorithms – although it can be said that there is a large overlap between these classes. We will also give a number of other algorithms that solve a slightly different problem.

5.1 Matching Pursuit

Matching pursuit is a class of iterative algorithms that decomposes a signal into a linear expansion of functions that form a dictionary. Matching pursuit was first introduced in (Mallat and Zhang 1993) and is closely related to *Projection Pursuit Regression* as in (Friedman and Stuetzle 1981). At each iteration of the algorithm, matching pursuit chooses dictionary elements in a greedy fashion that best approximate the signal.

We will now describe in more detail the matching pursuit process. Let $\mathscr{H}$ be a Hilbert space and let $\mathscr{D} = \{g_\gamma\}_{\gamma \in \Gamma}$ be a family of vectors where $\|g_\gamma\| = 1$. We say that $\mathscr{D}$ is a *dictionary*. Let $\mathbb{V}$ be the closed linear span of the dictionary and note that the finite linear combinations of dictionary elements are dense in $\mathbb{V}$. We say that the dictionary is *complete* if and only if $\mathbb{V} = \mathscr{H}$.

Let $f \in \mathscr{H}$, then we compute a linear expansion of f over the dictionary $\mathscr{D}$ by successive approximation. For any vector $g \in \mathscr{D}$ we can decompose f by writing

$$f = \langle f, g \rangle \, g + r^{(g)}, \tag{5.1}$$

where $r^{(g)}$ is the residual after approximating f by g. As the residual is orthogonal to f we have

$$\|f\|^2 = |\langle f, g \rangle|^2 + \left\| r^{(g)} \right\|^2. \tag{5.2}$$

For the "best" approximation we want to minimise the norm of the residual, hence we want $|\langle f, g \rangle|^2$ maximal. As it is not always possible to choose such a vector (in the case of infinite dimensionality), it suffices to take a dictionary element that is "almost best" in the sense that

$$|\langle f, g \rangle|^2 \geqslant \alpha \sup_{\gamma \in \Gamma} |\langle f, g_\gamma \rangle|, \tag{5.3}$$

for some α that satisfies $0 < \alpha \leqslant 1$, which is guaranteed to exist by the Axiom of Choice (Zermelo 1904). Fortunately however, we are only interested in finite dimensional Hilbert spaces and dictionaries, so we do not need to be concerned by this.

At each iteration of the matching pursuit algorithm we choose the almost best approximation for the current residual from the dictionary. Suppose that we have carried out n iterations so we have the n-th order residual $r^{(n)}$. We then choose the function $g^{(n+1)}$ from $\mathscr{D}$ that is almost best in the sense of Equation (5.3), thus decomposing $r^{(n)}$ into

$$r^{(n)} = \left\langle r^{(n)}, g^{(n)} \right\rangle g^{(n)} + r^{(n+1)}. \tag{5.4}$$

Expanding this decomposition up until the original function f we get

$$f = \sum_{i=0}^{n} \left\langle r^{(i)}, g^{(i)} \right\rangle g^{(i)} + r^{(n+1)}. \tag{5.5}$$

As the residual $r^{(i+1)}$ is always orthogonal to $r^{(i)}$ we get the energy conservation result

$$\|f\|^2 = \sum_{i=0}^{n} \left| \left\langle f, g^{(i)} \right\rangle \right|^2 + \left\| r^{(n+1)} \right\|^2. \tag{5.6}$$

It can be shown that matching pursuit completely recovers the components of f that are explained by the dictionary elements (Mallat and Zhang 1993).

5.1.1 Orthogonal Matching Pursuit

Orthogonal matching pursuit (OMP) is an improvement on matching pursuit. The principle is the same, at every iteration an element is picked from the dictionary that best approximates the residual. However rather than simply taking the scalar product of the residual and the new dictionary element to get the coefficient weight, we fit the original function to all the already selected dictionary elements via least squares or projecting the function orthogonally onto all selected dictionary atoms, hence the term orthogonal matching pursuit. (Pati, Rezaiifar, and Krishnaprasad 1993; Mallat, Davis, and Zhang 1994; Davis, Mallat, and Avellaneda 1997)

Orthogonal matching pursuit has been successfully used for signal recovery, however many complaints appeared concerning its performance in compressive sensing, such as (DeVore and Temlyakov 1996). More recent work though in (Tropp and Gilbert 2007) suggests that OMP can indeed perform well in the compressive sensing arena. The recent paper (Tropp and Gilbert 2007) contains two main theorems concerning the application of orthogonal matching pursuit

(see Algorithm 5.1) to compressed sensing. Consider the task of finding the k sparse signal $\mathbf{x} \in \mathbb{R}^n$ given the measurements $\mathbf{y} = \Phi\mathbf{x}$ and the measurement matrix $\Phi \in \mathbb{R}^{m \times n}$.

Theorem 5.1 (Theorem 2 of (Tropp and Gilbert 2007)). *Fix $\delta \in (0, 0.36)$ and choose $m \geqslant C\, k \log\left(\frac{n}{\delta}\right)$. Assume that $\mathbf{x} \in \mathbb{R}^n$ is k-sparse and let $\Phi \in \mathbb{R}^{m \times n}$ have i.i.d. entries from the Gaussian distribution $\mathcal{N}\left(0, \frac{1}{m}\right)$. Then, given the data $\mathbf{y} = \Phi\mathbf{x}$, orthogonal matching pursuit (Algorithm 5.1) can reconstruct the signal $\mathbf{x}$ with probability exceeding $1 - 2\delta$ where the constant C satisfies $C \leqslant 20$. For large k it can be shown that $C \approx 4$.*

Theorem 5.2 (Theorem 6 of (Tropp and Gilbert 2007)). *Fix $\delta \in (0, 0.36)$ and choose $m \geqslant C\, k \log\left(\frac{n}{\delta}\right)$ and suppose that $\mathbf{x} \in \mathbb{R}^n$ is a k-sparse signal. Let Φ be an admissible measurement matrix. Then, given the data $\mathbf{y} = \Phi\mathbf{x}$, orthogonal matching pursuit can reconstruct the signal with probability exceeding $1 - \delta$.*

For Theorem 5.2 to make sense we need to define what an admissible measurement matrix is.

Definition 5.3. *An* admissible measurement matrix *$\Phi \in \mathbb{R}^{m \times n}$ for recovering k-sparse signals is a matrix with the following properties.*

(a) ***Independence:*** *The columns of Φ are statistically independent.*

(b) ***Normalisation:*** *$\mathbb{E}\left[\phi_j\right] = 1$ for $j = 1, \ldots, n$, where the ϕ_j are the columns of Φ.*

(c) ***Joint Correlation:*** *For k vectors $\{\mathbf{u}_i\}$ such that $\|\mathbf{u}_i\|_2 \leqslant 1$ and for every column ϕ_j of Φ,*

$$\mathbb{P}\left[\max_i \left|\langle \phi_j, \mathbf{u}_i \rangle\right| \leqslant \varepsilon\right] \geqslant 1 - 2ke^{-c\varepsilon^2 m}.$$

(d) ***Smallest Singular Value:*** *For any submatrix Φ' of Φ with $\Phi' \in \mathbb{R}^{m \times k}$ the smallest singular value $\sigma_k(\Phi')$ satisfies*

$$\mathbb{P}\left[\sigma_k(\Phi') \geqslant \frac{1}{2}\right] \geqslant 1 - e^{-cm}.$$

The running time of the OMP algorithm is dominated by step 4 of Algorithm 5.1, finding the maximally correlated column of Φ with the residual. By maintaining a QR factorisation of $\Phi^{(i)}$ the total running time for this algorithm is $\mathcal{O}\left(kmn\right)$, (Tropp and Gilbert 2007).

We can now prove Theorem 5.2.

Proof of Theorem 5.2. Without loss of generality, assume that the data vector $\mathbf{x} \in \mathbb{R}^n$ is k-sparse and that the first k components of $\mathbf{x}$ are the non-zero components. Then the observation vector $\mathbf{y} \in \mathbb{R}^m$ is a linear combination of the first k columns of Φ. To this end partition Φ into two matrices $\Phi = \left[\Phi_{opt} \,|\, \Phi_0\right]$ where Φ_{opt} has k columns.

We are interested in the event that the algorithm correctly identifies $\mathbf{x}$ after i iterations, so let $\mathcal{E}_{success}$ denote this event. Furthermore let $\mathcal{E}_\sigma$ denote the event that smallest singular value of Φ_{opt} is at least equal to $\frac{1}{2}$, i.e.

$$\mathcal{E}_\sigma \triangleq \left\{\sigma_{min}(\Phi_{opt}) \geqslant \frac{1}{2}\right\}. \tag{5.7}$$

We then have

$$\mathbb{P}\left[\mathcal{E}_{success}\right] \geqslant \mathbb{P}\left[\mathcal{E}_{success} \cap \mathcal{E}_\sigma\right] = \mathbb{P}\left[\mathcal{E}_{success} | \mathcal{E}_\sigma\right] \cdot \mathbb{P}\left[\mathcal{E}_\sigma\right]. \tag{5.8}$$

Algorithm 5.1 Orthogonal Matching Pursuit

Input:

- ▶ A measurement matrix $\Phi \in \mathbb{R}^{m \times n}$.
- ▶ Observation vector $\mathbf{y} \in \mathbb{R}^m$.
- ▶ Sparsity level k of the ideal signal $\mathbf{x} \in \mathbb{R}^n$.

Output:

- ▶ An estimate $\hat{\mathbf{x}} \in \mathbb{R}^n$ of the ideal signal $\mathbf{x}$.
- ▶ A set Λ_k containing the positions of the non-zero elements of $\hat{\mathbf{x}}$.
- ▶ An approximation to the measurements $\mathbf{y}$ by $\mathbf{a}_k$.
- ▶ The residual $\mathbf{r} = \mathbf{y} - \mathbf{a}_k$.

1: $\mathbf{r}^{(0)} \leftarrow \mathbf{y}$ ▷ Initialise the residual
2: $\Lambda^{(0)} \leftarrow \varnothing$ ▷ Initialise the indices
3: **for** $i = 1, \dots, k$ **do**
4:     $\lambda^{(i)} \leftarrow \arg\max_{j=1,\dots,n} \left| \left\langle \mathbf{r}^{(i-1)}, \phi_j \right\rangle \right|$ ▷ The column of Φ that is most correlated with $\mathbf{r}_{i-1}$
5:     $\Lambda^{(i)} \leftarrow \Lambda^{(i-1)} \cup \lambda^{(i)}$
6:     $\Phi^{(i)} \leftarrow \left[\Phi^{(i-1)} \; \phi_{\lambda^{(i)}} \right]$
7:     $\mathbf{x}^{(i)} \leftarrow \arg\min_{\hat{\mathbf{x}}} \left\| \mathbf{y} - \Phi^{(i)} \hat{\mathbf{x}} \right\|_2$ ▷ Solve the Least Squares for new signal estimate
8:     $\mathbf{a}^{(i)} \leftarrow \Phi^{(i)} \mathbf{x}^{(i)}$ ▷ New data approximation
9:     $\mathbf{r}^{(i)} \leftarrow \mathbf{y} - \mathbf{a}^{(i)}$ ▷ New residual
10: **end for**
11: $\hat{\mathbf{x}} \leftarrow \mathbf{x}^{(k)}$
12: **return** $\hat{\mathbf{x}}, \Lambda^{(k)}, \mathbf{a}^{(k)}, \mathbf{r}^{(k)}$

By the definition of an admissible measurement matrix we know that $\mathbb{P}\left[\mathcal{E}_\sigma\right] \geqslant 1 - e^{-cm}$. It suffices to check that the algorithm correctly identifies the support of $\mathbf{x}$, as the condition on the singular values of Φ_{opt} ensures that this matrix is invertible and hence solving the least squares problem on this support will return the correct values.

Define the *greedy selection ratio*, $\rho(\mathbf{r})$, which was studied in (Tropp 2004)

$$\rho(\mathbf{r}) \triangleq \frac{\left\| \Phi_0^\mathsf{T} \mathbf{r} \right\|_\infty}{\left\| \Phi_{opt}^\mathsf{T} \mathbf{r} \right\|_\infty} = \frac{\max_\psi |\langle \psi, \mathbf{r} \rangle|}{\left\| \Phi_{opt}^\mathsf{T} \mathbf{r} \right\|_\infty}, \tag{5.9}$$

where the maximisation runs over the columns of Φ. If $\mathbf{r}$ is the residual appearing in the OMP algorithm then a value of $\rho(\mathbf{r}) < 1$ means that the algorithm selects a column from Φ_{opt}. If $\rho(\mathbf{r}) = 1$ a column from Φ_{opt} and Φ_0 both achieve the maximum correlation with the residual and it is not clear which column the algorithm will then pick.

Consider the scenario now, where we know the support of $\mathbf{x}$ and we then run the OMP algorithm with the restricted measurement matrix Φ_{opt}, producing a series of residuals $\mathbf{q}^{(k)}$ and column indices $\omega^{(k)}$. We will compare this to the actual invocation of the algorithm producing the residuals $\mathbf{r}^{(k)}$ and selected columns $\lambda^{(k)}$. We will show that if the imaginary execution at iteration k has $\rho(\mathbf{q}^{(k)}) < 1$ then we must have $\rho(\mathbf{r}^{(k)}) < 1$.

Initially we have $\mathbf{r}^{(0)} = \mathbf{y}$ and $\mathbf{q}^{(0)} = \mathbf{y}$, hence in the base case if $\rho(\mathbf{q}^{(0)}) < 1$ we have $\rho(\mathbf{r}^{(0)}) < 1$ and thus the algorithm must select the same columns $\lambda^{(0)} = \omega^{(0)}$. We complete the proof of

this property by induction. Assume then that for the first k iterations the two invocations of the algorithm produce the same set of column indices $\lambda^{(i)} = \omega^{(i)}$ for $0 \leqslant i \leqslant k$ and hence the same residuals, in particular $\mathbf{q}^{(k)} = \mathbf{r}^{(k)}$. Therefore if $\rho(\mathbf{q}^{(k+1)}) < 1$ we have $\rho(\mathbf{r}^{(k+1)}) < 1$ and thus the algorithms again select the same column, i.e. $\lambda^{(k+1)} = \omega^{(k+1)}$. Note that the residuals $\mathbf{q}^{(i)}$ lie in the column span of Φ_{opt} and as such are statistically independent from the matrix Φ_0 as the columns of the full matrix Φ are independent. Therefore we conclude that

$$\mathbb{P}\left[\mathcal{E}_{success}\middle|\mathcal{E}_\sigma\right] \geqslant \mathbb{P}\left[\max_{i=1,\dots,s} \rho\left(\mathbf{q}^{(i)}\right) < 1 \middle| \mathcal{E}_\sigma\right]. \tag{5.10}$$

Assume then that the smallest singular value of Φ_{opt} is at least $\frac{1}{2}$, i.e. that the event $\mathcal{E}_\sigma$ occurs. Observe that for all s-sparse vectors $\hat{\mathbf{x}}$ we have $\|\hat{\mathbf{x}}\|_2 \leqslant \sqrt{s}\,\|\hat{\mathbf{x}}\|_\infty$. Therefore at each iteration i of the algorithm we have

$$\rho\left(\mathbf{q}^{(i)}\right) = \frac{\max_\phi \left|\left\langle \psi, \mathbf{q}^{(i)}\right\rangle\right|}{\left\|\Phi_{opt}^\mathsf{T}\mathbf{q}^{(i)}\right\|_\infty} \tag{5.11a}$$

$$\leqslant \frac{\sqrt{s}\max_\phi \left|\left\langle \psi, \mathbf{q}^{(i)}\right\rangle\right|}{\left\|\Phi_{opt}^\mathsf{T}\mathbf{q}^{(i)}\right\|_2}, \tag{5.11b}$$

For convenience define the vector

$$\mathbf{u}^{(i)} \triangleq \frac{\mathbf{q}^{(i)}}{2\left\|\Phi_{opt}^\mathsf{T}\mathbf{q}^{(i)}\right\|_2}. \tag{5.12}$$

Since the Rayleigh quotient of a matrix is bounded by the square root of the maximum and minimum singular values we have

$$\frac{\left\|\Phi_{opt}^\mathsf{T}\mathbf{q}^{(i)}\right\|_2}{\left\|\mathbf{q}^{(i)}\right\|_2} \geqslant \sigma_s\left(\Phi_{opt}\right) \geqslant \frac{1}{2}. \tag{5.13}$$

This gives us

$$\rho\left(\mathbf{q}^{(i)}\right) \leqslant 2\sqrt{s}\max_\phi \left|\left\langle \phi, \mathbf{u}^{(i)}\right\rangle\right|. \tag{5.14}$$

We can use this to create a lower bound for the probability that all the $\rho(\mathbf{q}^{(i)})$ are less than 1

$$\mathbb{P}\left[\max_{i=1,\dots,s} \rho\left(\mathbf{q}^{(i)}\right) < 1 \middle| \mathcal{E}_\sigma\right] \geqslant \mathbb{P}\left[\max_i \max_\phi \left|\left\langle \phi, \mathbf{u}^{(i)}\right\rangle\right| < \frac{1}{2\sqrt{s}} \middle| \mathcal{E}_\sigma\right] \tag{5.15a}$$

$$\geqslant \prod_\phi \mathbb{P}\left[\max_i \left|\left\langle \phi, \mathbf{u}^{(i)}\right\rangle\right| < \frac{1}{2\sqrt{s}} \middle| \mathcal{E}_\sigma\right], \tag{5.15b}$$

by swapping the two max terms and by independence of the columns of Φ_0. Then by the joint correlation property of admissible measurement matrices and the fact that $\mathbf{u}^{(i)}$ is independent of the columns of Φ_0, we have a lower bound on each of the product terms appearing above

$$\mathbb{P}\left[\max_j \left|\left\langle \varphi, \mathbf{u}^{(i)}\right\rangle\right| \leqslant \varepsilon\right] \geqslant 1 - 2se^{-c\varepsilon^2 m}, \tag{5.16}$$

which implies that

$$\mathbb{P}\left[\max_{i=1,\ldots,s}\rho\left(\mathbf{q}^{(i)}\right)<1\,\Big|\,\mathcal{E}_\sigma\right]\geqslant\left(1-2se^{-cm/4s}\right)^{n-s},\tag{5.17}$$

as there are $n-s$ columns in Φ_0. Combining this with the previous property that the singular values of Φ_{opt} are lower bounded we now can write

$$\mathbb{P}\left[\mathcal{E}_{success}\right]\geqslant\left(1-2se^{-cm/4s}\right)^{n-s}\left(1-e^{-cm}\right)\tag{5.18a}$$

$$\geqslant 1-2s(n-s)e^{-cm/4s}-e^{-cm},\tag{5.18b}$$

where we use the Taylor expansion $(1-x)^k\geqslant 1-kx$ for $x\leqslant 1$ and $k\geqslant 1$. As the sparsity s is less than $n/2$ we have $s(n-s)\leqslant n^2/4$, so we can rewrite the above equation to get

$$\mathbb{P}\left[\mathcal{E}_{success}\right]\geqslant 1-n^2 e^{-\tilde{c}m/s}.\tag{5.19}$$

If we ensure that $m\geqslant C\cdot s\log(n/\delta)$, then the probability of failure is less than δ. The requirement that $\delta<0.36$ ensures that the logarithm term is always larger than 1. $\qquad\square$

5.1.2 Stagewise Orthogonal Matching Pursuit

Stagewise orthogonal matching pursuit or StOMP, introduced in (Donoho, Tsaig, Drori, and Starck 2007), is an improvement on the OMP algorithm presented in the previous section. In contrast to OMP it allows multiple coefficients to be added to the model in a single iteration and runs for a fixed number of iterations. We list StOMP as Algorithm 5.2.

Algorithm 5.2 StOMP

Input:

- ▶ Number of iterations S to perform
- ▶ Threshold value t_s
- ▶ $\mathbf{y}\in\mathbb{R}^m$ and the measurement matrix $\Phi\in\mathbb{R}^{m\times n}$

Output:

- ▶ $\hat{\mathbf{x}}\in\mathbb{R}^n$ such that $\hat{\mathbf{x}}$ is s-sparse and $\mathbf{y}=\Phi\hat{\mathbf{x}}$

1: $\mathbf{x}^{(0)}\leftarrow\mathbf{0}$
2: $\Lambda^{(0)}\leftarrow\varnothing$
3: **for** $i=1,\ldots,S$ **do**
4: $\quad\mathbf{r}^{(i)}\leftarrow\mathbf{y}-\Phi\mathbf{x}^{(i-1)}$ $\hfill\triangleright$ Residual
5: $\quad\mathbf{v}^{(i)}\leftarrow\Phi^{\mathsf{T}}\mathbf{r}^{(i)}$ $\hfill\triangleright$ Matched filter
6: $\quad\Gamma^{(i)}\leftarrow\left\{j:\left|v_j^{(i)}\right|>t_s\right\}$ $\hfill\triangleright$ Select components larger than the threshold t_s
7: $\quad\Lambda^{(i)}\leftarrow\Lambda^{(i-1)}\cup\Gamma^{(i)}$
8: $\quad\mathbf{x}^{(i)}\leftarrow\Phi_{\Lambda^{(i)}}^{\dagger}\mathbf{y}$ $\hfill\triangleright$ Projection
9: **end for**
10: **return** $\hat{\mathbf{x}}\leftarrow\mathbf{x}^{(S)}$

The choice of the thresholding parameter t_s is inspired Gaussian noise removal, such as arising in digital communications, known as *Multiple Access Interference* or MAI (Verdú 1998). In this problem the MAI typically has Gaussian behaviour, in particular if the measurement matrix

$\Phi \in \mathbb{R}^{m \times n}$ has columns sampled from the unit sphere and m, n are large, then the entries $\mathbf{z} = \Phi^\top \mathbf{y} - \mathbf{x} = \Phi^\top \Phi \mathbf{x} - \mathbf{x}$ has a histogram which is nearly Gaussian and has standard deviation

$$\sigma \approx \frac{\|\mathbf{x}\|_2}{\sqrt{m}}. \tag{5.20}$$

For the justification of this, see (Donoho, Tsaig, Drori, and Starck 2007).

Let Γ be the support of $\mathbf{x}$ and let Γ_S be the support of $\hat{\mathbf{x}} = \mathbf{x}_S$, the output of Algorithm 5.2. The components of Γ_S we call *discoveries* and the components in $\Gamma_S \setminus \Gamma$ are called *false discoveries*. Then if a component appears in Γ but not in Γ_S, we call this a *missed detection*. If a component occurs in Γ_S but not in Γ, we call this a *false alarm*. Observe that a false alarm is a false discovery, but we distinguish between them when calculating the rate at which these occur. The false discovery rate is the fraction of false discoveries in Γ_S and the false alarm rate is the number of false alarms over the number of components not in Γ.

The process for choosing the threshold parameter is then governed by one of two strategies:

   (a) **False Alarm Control.** We attempt to guarantee that the number of false alarms, occurring across all iterations does not exceed $m - s$. The threshold is chosen so the false alarm rate does not exceed a specific per-iteration amount.

   (b) **False Discovery Control.** The threshold is chosen so as not to exceed a certain fraction of the total number of components added across all iterations.

These two strategies are discussed further in (Abramovich, Benjamini, Donoho, and Johnstone 2000). Software implementing this algorithm can be found in the *SparseLab* MATLAB software package (Donoho, Drori, Stodden, Tsaig, and Shahram 2007).

5.1.3   Gradient Pursuit

Gradient pursuit (GP) is yet another variation of matching pursuit. Instead of taking the update to simply be the scalar-product of the residual and dictionary element, the update occurs in a particular direction (Blumensath and Davies 2008). Once again consider our usual scenario, given the matrix $\Phi \in \mathbb{R}^{m \times n}$ where $m < n$ and given the observations $\mathbf{y} = \Phi \mathbf{x}$ for some unknown but sparse vector $\mathbf{x} \in \mathbb{R}^n$ we wish to reconstruct the vector $\mathbf{x}$. In gradient pursuit, at iteration k, the following update to $\mathbf{x}^{(k-1)}$ occurs

$$\mathbf{x}^{(k)} = \mathbf{x}^{(k-1)} + a^{(k)} \mathbf{d}^{(k)}, \tag{5.21}$$

where $\mathbf{d}^{(k)}$ is the update direction. Once a direction has been selected, the optimal step-size $a^{(n)}$ (in terms of minimising the squared error $\left\| \mathbf{y} - \Phi \mathbf{x}^{(k)} \right\|_2^2$), is shown in (Golub and Loan 1996, pp. 521) to be

$$a^{(k)} = \frac{\left\langle \mathbf{r}^{(k)}, \Phi_{\Gamma^{(k)}} \mathbf{d}^{(k)} \right\rangle}{\left\| \Phi_{\Gamma^{(k)}} \mathbf{d}^{(k)} \right\|_2^2}, \tag{5.22}$$

In matching pursuit and orthogonal matching pursuit, the update direction is taken to be in the direction of the element in the dictionary $\mathscr{D}$ that has largest inner product with the current residual, $\mathbf{r}^{(k)}$. In OMP, once added, an atom will not be selected again as the process of or-

thogonalisation ensures that all future residuals will remain orthogonal to all currently selected atoms. In MP and GP however, orthogonality is not ensured. For this reason we allow the GP algorithm to use an update direction that may have already been used.

Algorithm 5.3 Gradient Pursuit

Input:

▶ $\mathbf{y} \in \mathbb{R}^m$ and the measurement matrix $\Phi \in \mathbb{R}^{m \times n}$

Output:

▶ $\hat{\mathbf{x}} \in \mathbb{R}^n$ such that $\hat{\mathbf{x}}$ is sparse and $\mathbf{y} = \Phi\hat{\mathbf{x}}$

1: $\mathbf{x}^{(0)} \leftarrow \mathbf{0}$
2: $\mathbf{r}^{(0)} \leftarrow \mathbf{y}$
3: $\Gamma \leftarrow \varnothing$
4: $k \leftarrow 0$
5: **while** Halting condition false **do**
6:      $k \leftarrow k + 1$
7:      $\mathbf{g}^{(k)} \leftarrow \Phi^\mathsf{T} \mathbf{r}^{(k-1)}$
8:      $\lambda^{(k)} \leftarrow \arg\max_j \left| g_j^{(k)} \right|$ ▷ Select largest component of $\mathbf{g}^{(k)}$
9:      $\Gamma^{(k)} \leftarrow \Gamma^{(k-1)} \cup \lambda^{(k)}$
10:     **if** Gradient Pursuit **then**   ▷ Calculate update direction $\mathbf{d}^{(k)}$
11:          $\mathbf{d}^{(k)} \leftarrow \Phi_{\Gamma^{(k)}}^\mathsf{T} \left(\mathbf{y} - \Phi_{\Gamma^{(k)}} \mathbf{x}_{\Gamma^{(k)}}^{(k-1)} \right)$
12:     **else if** Conjugate Gradient Pursuit **then**
13:          $\mathbf{g}^{(k)} \leftarrow \Phi_{\Gamma^{(k)}}^\mathsf{T} \left(\mathbf{y} - \Phi_{\Gamma^{(k)}} \mathbf{x}_{\Gamma^{(k)}}^{(k-1)} \right)$
14:          $\mathbf{b} \leftarrow - \left(\left(\mathbf{D}^{(k-1)} \right)^\mathsf{T} \mathbf{G} \mathbf{D}^{(k-1)} \right)^{-1} \left(\left(\mathbf{D}^{(k-1)} \right)^\mathsf{T} \mathbf{G} \mathbf{g}^{(k-1)} \right)$
15:          $\mathbf{d}^{(k)} \leftarrow \mathbf{g}^{(k)} + \mathbf{D}^{(k-1)} \mathbf{b},$
16:     **else if** Approximate Conjugate Gradient Pursuit **then**
17:          $\mathbf{g}^{(k)} \leftarrow \Phi_{\Gamma^{(k)}}^\mathsf{T} \left(\mathbf{y} - \Phi_{\Gamma^{(k)}} \mathbf{x}_{\Gamma^{(k)}}^{(k-1)} \right)$
18:          $b \leftarrow - \dfrac{\left\langle \left(\Phi_{\Gamma^{(k)}} \mathbf{d}^{(k-1)} \right), \left(\Phi_{\Gamma^{(k)}} \mathbf{g}^{(k)} \right) \right\rangle}{\left\| \Phi_{\Gamma^{(k)}} \mathbf{d}^{(k-1)} \right\|_2^2}$
19:          $\mathbf{d}^{(k)} \leftarrow \mathbf{g}^{(k)} + \mathbf{d}^{(k-1)} b$
20:     **end if**
21:     $a^{(k)} \leftarrow \dfrac{\left\langle \mathbf{r}^{(k)}, \Phi_{\Gamma^{(k)}} \mathbf{d}^{(k)} \right\rangle}{\left\| \Phi_{\Gamma^{(k)}} \mathbf{d}^{(k)} \right\|_2^2}$
22:     $\mathbf{x}_{\Gamma^{(k)}}^{(k)} \leftarrow \mathbf{x}_{\Gamma^{(k-1)}}^{(k-1)} + a^{(k)} \mathbf{d}^{(k)}$
23:     $\mathbf{r}^{(k)} \leftarrow \mathbf{r}^{(k-1)} - a^{(k)} \Phi_{\Gamma^{(k)}} \mathbf{d}^{(k)}$
24: **end while**
25: $\hat{\mathbf{x}} \leftarrow \mathbf{x}^{(k)}$
26: **return** $\hat{\mathbf{x}}$

We will discuss three different methods for calculating the update direction in step 10 of Algorithm 5.3.

(a) **Gradient Pursuit** The first option is to use the direction that minimises $\left\| \mathbf{y} - \Phi \mathbf{x}^{(k-1)} \right\|_2$. Hence the update direction is simply

$$\mathbf{d}^{(k)} = \Phi_{\Gamma^{(k)}}^\mathsf{T} \left(\mathbf{y} - \Phi_{\Gamma^{(k)}} \mathbf{x}_{\Gamma^{(k)}}^{(k-1)} \right), \tag{5.23}$$

which is the same as the classical matching pursuit algorithm. The authors of (Blumensath and Davies 2008) simply call this the "Gradient Pursuit method".

(b) **Conjugate Gradient Pursuit** Another popular update method for iteratively solving equations is the conjugate gradient method (Golub and Loan 1996, §10.2). Briefly, if $\varphi(\mathbf{x}) = \frac{1}{2}\mathbf{x}^\mathsf{T}\mathbf{G}\mathbf{x} - \mathbf{b}^\mathsf{T}\mathbf{x}$ is the cost function to be minimised (which is equivalent to solving $\mathbf{G}\mathbf{x} = \mathbf{b}$ for $\mathbf{x}$), then the conjugate gradient method chooses a directional update $\mathbf{d}^{(k)}$ that is $\mathbf{G}$-conjugate to all the previous directions, in other words

$$\mathbf{d}^{(k)}\mathbf{G}\mathbf{d}^{(i)} = 0, \qquad \forall i < k. \tag{5.24}$$

In our case we take $\mathbf{G} = \Phi_{\Gamma^{(k)}}^\mathsf{T}\Phi_{\Gamma^{(k)}}$, which means we are minimising the expression

$$\left\| \mathbf{y} - \Phi_{\Gamma^{(k)}}\mathbf{x}_{\Gamma^{(k)}}^{(k)} \right\|_2^2. \tag{5.25}$$

Let $\mathbf{D}^{(i)}$ be the matrix whose columns are the update directions for the first i iterations and let $\mathbf{g}^{(i)}$ be the gradient of the the cost function in iteration i. Then the new update direction in iteration k is given by

$$\mathbf{d}^{(k)} = \mathbf{g}^{(k)} + \mathbf{D}^{(k-1)}\mathbf{b}, \tag{5.26a}$$

$$\text{where} \quad \mathbf{b} = -\left(\left(\mathbf{D}^{(k-1)}\right)^\mathsf{T} \mathbf{G}\mathbf{D}^{(k-1)} \right)^{-1} \left(\left(\mathbf{D}^{(k-1)}\right)^\mathsf{T} \mathbf{G}\mathbf{g}^{(k-1)} \right). \tag{5.26b}$$

For more details and an efficient implementation for the matrix arithmetic, we refer you to (Blumensath and Davies 2008, pp. 3-5). Note that this method is similar to OMP . The difference is that OMP uses a full conjugate gradient solver at every iteration but for this proposed method, only a single update occurs for every new added element. In (Pati, Rezaiifar, and Krishnaprasad 1993) an OMP implementation is proposed using directional updates, but this requires matrix inversions. In contrast, the matrix "inversion" in Equation (5.26b) does not requires a full matrix inversion as $\left(\left(\mathbf{D}^{(k-1)}\right)^\mathsf{T} \mathbf{G}\mathbf{D}^{(k-1)} \right)$ is in fact diagonal, a consequence of the $\mathbf{G}$-conjugacy.

(c) **Approximate Conjugate Gradient Pursuit** Instead of calculating the exact conjugate gradient direction, here we simply approximate it, resulting in an approximation to the OMP algorithm. The advantage in this is speed and storage requirements. The approximation occurs in that we calculate a new direction that is conjugate to only a *subset* of all the previous update directions. Here we consider only a direction that is conjugate to the previous direction, but this can be extended to a larger number directions. This gives us

$$\mathbf{d}^{(k)} = \mathbf{g}^{(k)} + \mathbf{d}^{(k-1)}b. \tag{5.27}$$

Then $\mathbf{G}$-conjugacy implies that

$$\left\langle \left(\mathbf{G}\mathbf{d}^{(k-1)}\right), \left(\mathbf{g}^{(k)} + b\mathbf{d}^{(k-1)}\right) \right\rangle = 0, \tag{5.28}$$

which gives us

$$b = -\frac{\left\langle \left(\Phi_{\Gamma^{(k)}}\mathbf{d}^{(k-1)}\right), \left(\Phi_{\Gamma^{(k)}}\mathbf{g}^{(k)}\right) \right\rangle}{\left\| \Phi_{\Gamma^{(k)}}\mathbf{d}^{(k-1)} \right\|_2^2}. \tag{5.29}$$

These methods appear very similar to some other algorithms discussed in this thesis, in par-

ticular IHT (§5.2.1) and our MFR algorithm (Chapter 6). All of these algorithms have the same update direction, but the big difference is in how the sparsity constraint is enforced. For the gradient pursuit algorithms, a new dictionary element is added at every iteration, and once added, cannot be removed. In contrast, IHT and MFR make use of a pruning step, so at every iteration we keep only the most important (decided by the largest magnitude) dictionary elements, thus elements can be both added and removed.

5.1.4 CoSaMP

An extension to orthogonal matching pursuit algorithms is the CoSaMP (COmpressive SAmpling Matching Pursuit) algorithm published in (Needell and Tropp 2008). The basis of the algorithm is OMP but CoSaMP, Algorithm 5.4, can be shown to have tighter bounds on its convergence and performance, as in Theorem 5.4.

Theorem 5.4 (CoSaMP, Theorem A in (Needell and Tropp 2008)). *Let $\Phi \in \mathbb{R}^{m \times n}$ be a measurement matrix that obeys the restricted isometry property of order $2s$ obeying $\delta_{2s} \leqslant c$. Let $\mathbf{y} = \Phi\mathbf{x} + \mathbf{e}$ be a measurement of the signal $\mathbf{x} \in \mathbb{R}^n$ with error $\mathbf{e} \in \mathbb{R}^n$ of arbitrary noise. Then for a given precision parameter η, the CoSaMP algorithm, Algorithm 5.4, produces an s-sparse vector $\hat{\mathbf{x}}$ that satisfies*

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_2 \leqslant C \cdot \max\left\{\eta, \frac{1}{\sqrt{s}}\left\|\mathbf{x} - \mathbf{x}^{s/2}\right\|_1 + \|\mathbf{e}\|_2\right\}, \tag{5.30}$$

where $\mathbf{x}^{s/2}$ is a best $s/2$-sparse approximation to $\mathbf{x}$. Furthermore the algorithm has running time $\mathcal{O}\left(\mathcal{L} \cdot \log\left(\frac{\|\mathbf{x}\|_2}{\eta}\right)\right)$ where $\mathcal{L}$ is the cost of a matrix multiply with Φ or Φ^{T}.

Algorithm 5.4 CoSaMP

Input:

- ▶ s the sparsity of $\mathbf{x}$
- ▶ $\mathbf{y} \in \mathbb{R}^m$ and the measurement matrix $\Phi \in \mathbb{R}^{m \times n}$

Output:

- ▶ $\hat{\mathbf{x}} \in \mathbb{R}^n$ such that $\hat{\mathbf{x}}$ is s-sparse and $\mathbf{y} = \Phi\hat{\mathbf{x}}$

1: $\mathbf{x}^{(0)} \leftarrow \mathbf{0}$
2: $\mathbf{v} \leftarrow \mathbf{y}$
3: $k \leftarrow 0$
4: **while** Halting condition false **do**
5:      $k \leftarrow k + 1$
6:      $\mathbf{z} \leftarrow \Phi^{\mathsf{T}}\mathbf{v}$ ▷ Signal proxy
7:      $\Omega \leftarrow \operatorname{supp}\left(\mathbf{z}^{2s}\right)$ ▷ Support of best $2s$-sparse approximation "Identification"
8:      $\Gamma \leftarrow \Omega \cup \operatorname{supp}\left(\mathbf{x}^{(k-1)}\right)$ ▷ Merge supports
9:      $\bar{\mathbf{x}} \leftarrow \arg\min_{\tilde{\mathbf{x}}:\operatorname{supp}(\tilde{\mathbf{x}})=\Gamma} \|\Phi\tilde{\mathbf{x}} - \mathbf{y}\|_2$ ▷ Solve Least Squares
10:     $\mathbf{x}^{(k)} \leftarrow \bar{\mathbf{x}}^s$ ▷ Prune: best s-sparse approximation
11:     $\mathbf{v} \leftarrow \mathbf{y} - \Phi\mathbf{x}^{(k)}$ ▷ Update current sample
12: **end while**
13: $\hat{\mathbf{x}} \leftarrow \mathbf{x}^{(k)}$
14: **return** $\hat{\mathbf{x}}$

The CoSaMP consists of five main steps, each of which will be covered by a lemma analysing its performance.

(a) **Identification:** Finds the largest $2s$ components of the signal proxy as in line 7 of Algorithm 5.4, covered in Lemma 5.7.

(b) **Support Merge:** Merges the support of the signal proxy with the support of the solution from the previous iteration, line 8 of Algorithm 5.4, see Lemma 5.8.

(c) **Estimation:** Estimates a solution via least squares with the constraint that the solution lies on a particular support, line 9 of Algorithm 5.4, see Lemma 5.9.

(d) **Pruning:** Takes the solution estimate and compresses it to the required support, line 10 of Algorithm 5.4, see Lemma 5.10.

(e) **Sample Update:** Updates the "sample", namely the residual in Φ-space, line 11 of Algorithm 5.4.

To analyse the CoSaMP algorithm we first need to define the *unrecoverable energy* of a signal. Let

$$\nu \triangleq \|\mathbf{x} - \mathbf{x}^s\|_2 + \frac{1}{\sqrt{s}} \|\mathbf{x} - \mathbf{x}^s\|_1 + \|\mathbf{e}\|_2, \tag{5.31}$$

where $\mathbf{y} = \Phi\mathbf{x} + \mathbf{e}$ and $\mathbf{x}^s$ is a best s-sparse approximation to $\mathbf{x}$. This reflects the baseline error in the approximation $\hat{x}$ because of the noise in the signals and the fact that $\mathbf{x}$ is not necessarily sparse. We then have the following theorems.

Theorem 5.5 (Iteration Invariant, Theorem 2.1 of (Needell and Tropp 2008)). *For each k the signal approximation* $\mathbf{x}^{(k)}$ *is s-sparse and*

$$\left\| \mathbf{x} - \mathbf{x}^{(k+1)} \right\|_2 \leqslant \frac{1}{2} \left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2 + 10\nu. \tag{5.32}$$

In particular

$$\left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2 \leqslant 2^{-k} \|\mathbf{x}\|_2 + 20\nu. \tag{5.33}$$

Theorem 5.6 (Iteration Invariant: Sparse Case, Theorem 4.1 of (Needell and Tropp 2008)). *Assume that* $\mathbf{x}$ *is s-sparse, then for each* $k \geqslant 0$, *the signal approximation* $\mathbf{x}^{(k)}$ *is s-sparse and*

$$\left\| \mathbf{x} - \mathbf{x}^{(k+1)} \right\|_2 \leqslant \frac{1}{2} \left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2 + \frac{15}{2}\nu. \tag{5.34}$$

In particular

$$\left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2 \leqslant 2^{-k} \|\mathbf{x}\|_2 + 15\nu. \tag{5.35}$$

Following the proof in (Needell and Tropp 2008) we will first prove Theorem 5.6 and then remove the assumption of sparsity, proving Theorem 5.5. Let us define the residual in iteration k in the usual manner; $\mathbf{r}^{(k)} \triangleq \mathbf{x} - \mathbf{x}^{(k)}$. Recall that under the hypothesis of Theorem 5.6, $\mathbf{x}$ is s-sparse and the approximation is also s-sparse, hence $\mathbf{r}^{(k)}$ has at most $2s$ non-zero entries. The vector $\mathbf{v}$ from the algorithm of updated samples can be expressed in terms of the residual and an error $\mathbf{e}$

$$\mathbf{v}^{(k)} = \mathbf{y} - \Phi\mathbf{x} = \Phi\left(\mathbf{x} - \mathbf{x}^{(k)}\right) + \mathbf{e} = \Phi\mathbf{r}^{(k)} + \mathbf{e}. \tag{5.36}$$

Lemma 5.7 (Identification, Lemma 4.2 of (Needell and Tropp 2008)). *The set* $\Omega = \mathrm{supp}(\mathbf{z}_{2s})$, *where* $\mathbf{z} = \Phi^\mathsf{T}\mathbf{v}$, *contains at most 2s elements and*

$$\left\| \mathbf{r}^{(k)}_{\Omega^c} \right\|_2 \leqslant 0.2223 \left\| \mathbf{r}^{(k)} \right\|_2 + 2.34 \|\mathbf{e}\|_2. \tag{5.37}$$

Proof. Let $\Gamma = \text{supp}(\mathbf{r}^{(k)})$ which has size at most $2s$. As Ω is the support of the largest $2s$ elements in $\mathbf{z}$ we have $\|\mathbf{z}_\Gamma\|_2 \leqslant \|\mathbf{z}_\Omega\|_2$. Removing the components in $\Gamma \cap \Omega$ we get $\left\|\mathbf{z}_{\Gamma \setminus \Omega}\right\|_2 \leqslant \left\|\mathbf{z}_{\Omega \setminus \Gamma}\right\|_2$. These vectors have small support (less than $2s$) so we can apply the RIP constants. Then

$$\left\|\mathbf{z}_{\Omega \setminus \Gamma}\right\|_2 = \left\|\Phi_{\Omega \setminus \Gamma}^{\mathsf{T}} \mathbf{v}\right\|_2 \tag{5.38a}$$

$$= \left\|\Phi_{\Omega \setminus \Gamma}^{\mathsf{T}} \left(\Phi \mathbf{r}^{(k)} + \mathbf{e}\right)\right\|_2 \tag{5.38b}$$

$$\leqslant \left\|\Phi_{\Omega \setminus \Gamma}^{\mathsf{T}} \Phi \mathbf{r}^{(k)}\right\|_2 + \left\|\Phi_{\Omega \setminus \Gamma}^{\mathsf{T}} \mathbf{e}\right\|_2 \tag{5.38c}$$

$$\leqslant \delta_{4s} \left\|\mathbf{r}^{(k)}\right\|_2 + \sqrt{1 + \delta_{2s}} \|\mathbf{e}\|_2, \tag{5.38d}$$

where $\left\|\Phi_{\Omega \setminus \Gamma}^{\mathsf{T}} \Phi \mathbf{r}^{(k)}\right\|_2 \leqslant \delta_{4s} \left\|\mathbf{r}^{(k)}\right\|_2$ follows from Corollary 3.6 and $\left\|\Phi_{\Omega \setminus \Gamma}^{\mathsf{T}} \mathbf{e}\right\|_2 \leqslant \sqrt{1 + \delta_{2s}} \|\mathbf{e}\|_2$ follows from Proposition 3.4.

We apply a similar strategy to the term $\left\|\mathbf{z}_{\Gamma \setminus \Omega}\right\|_2$, again using Corollary 3.6 and Proposition 3.4 giving

$$\left\|\mathbf{z}_{\Gamma \setminus \Omega}\right\|_2 = \left\|\Phi_{\Gamma \setminus \Omega}^{\mathsf{T}} \mathbf{v}\right\|_2 \tag{5.39a}$$

$$= \left\|\Phi_{\Gamma \setminus \Omega}^{\mathsf{T}} \left(\Phi \mathbf{r}^{(k)} + \mathbf{e}\right)\right\|_2 \tag{5.39b}$$

$$\geqslant \left\|\Phi_{\Gamma \setminus \Omega}^{\mathsf{T}} \Phi \mathbf{r}_{\Gamma \setminus \Omega}^{(k)}\right\|_2 - \left\|\Phi_{\Gamma \setminus \Omega}^{\mathsf{T}} \Phi \mathbf{r}_\Omega^{(k)}\right\|_2 - \left\|\Phi_{\Gamma \setminus \Omega}^{\mathsf{T}} \mathbf{e}\right\|_2 \tag{5.39c}$$

$$\geqslant (1 - \delta_{2s}) \left\|\mathbf{r}_{\Gamma \setminus \Omega}^{(k)}\right\|_2 - \delta_{2s} \|\mathbf{r}\|_2 - \sqrt{1 + \delta_{2s}} \|\mathbf{e}\|_2. \tag{5.39d}$$

The fact that the residual is supported on Γ means that we can write $\mathbf{r}_{\Gamma \setminus \Omega}^{(k)} = \mathbf{r}_{\Omega^c}^{(k)}$. Combining this, Equation (5.38d) and Equation (5.39d) we get

$$(1 - \delta_{2s}) \left\|\mathbf{r}_{\Omega^c}^{(k)}\right\|_2 - \delta_{2s} \left\|\mathbf{r}^{(k)}\right\|_2 - \sqrt{1 + \delta_{2s}} \|\mathbf{e}\|_2 \leqslant \delta_{4s} \left\|\mathbf{r}^{(k)}\right\|_2 + \sqrt{1 + \delta_{2s}} \|\mathbf{e}\|_2 \tag{5.40a}$$

$$\iff \qquad \left\|\mathbf{r}_{\Omega^c}^{(k)}\right\|_2 \leqslant \frac{(\delta_{2s} + \delta_{4s}) \left\|\mathbf{r}^{(k)}\right\|_2 + 2\sqrt{1 + \delta_{2s}} \|\mathbf{e}\|_2}{1 - \delta_{2s}}. \tag{5.40b}$$

As $\delta_{2s} \leqslant \delta_{4s} \leqslant 0.1$ we get

$$\|\mathbf{r}_{\Omega^c}\|_2 \leqslant \frac{0.2 \left\|\mathbf{r}^{(k)}\right\|_2 + 2\sqrt{1.1} \|\mathbf{e}\|_2}{0.9} \leqslant 0.2223 \left\|\mathbf{r}^{(k)}\right\|_2 + 2.34 \|\mathbf{e}\|_2, \tag{5.41}$$

which is the desired result. $\qquad\square$

Lemma 5.8 (Support Merge, Lemma 4.3 of (Needell and Tropp 2008)). *Let Ω be a set of at most $2s$ indices, then the set $\Gamma = \Omega \cup \text{supp}(\mathbf{x}^{(k-1)})$ contains at most $3s$ elements and*

$$\|\mathbf{x}_{\Gamma^c}\|_2 \leqslant \|\mathbf{r}_{\Omega^c}\|_2. \tag{5.42}$$

Proof. As $\text{supp}(\mathbf{x}^{(k-1)}) \subset \Gamma$ and as $\Gamma^c \subset \Omega^c$ we have

$$\|\mathbf{x}_{\Gamma^c}\|_2 = \left\|(\mathbf{x} - \mathbf{x}^{(k-1)})_{\Gamma^c}\right\|_2 = \|\mathbf{r}_{\Gamma^c}\|_2 \leqslant \|\mathbf{r}_{\Omega^c}\|_2, \tag{5.43}$$

since $\mathbf{x}_{\Gamma^c}^{(k-1)} \equiv \mathbf{0}$, completing the proof of the lemma. $\qquad\square$

Lemma 5.9 (Least Squares Estimation, Lemma 4.4 of (Needell and Tropp 2008)). *Let Γ be a set of at most $3s$ indices and define $\bar{\mathbf{x}}$ to be the solution to the least squares problem*

$$\arg\min \|\Phi\hat{\mathbf{x}} - \mathbf{y}\|_2 \qquad such\ that \qquad \mathrm{supp}(\hat{\mathbf{x}}) = \Gamma, \tag{5.44}$$

with solution given by $\bar{\mathbf{x}}_\Gamma = \Phi_\Gamma^\dagger \mathbf{y}$ where $\mathbf{y} = \Phi\mathbf{x} + \mathbf{e}$. Then

$$\|\mathbf{x} - \bar{\mathbf{x}}\|_2 \leqslant 1.112 \|\mathbf{x}_{\Gamma^c}\|_2 + 1.06 \|\mathbf{e}\|_2. \tag{5.45}$$

Proof. Using the triangle inequality we have that

$$\|\mathbf{x} - \bar{\mathbf{x}}\|_2 \leqslant \|\mathbf{x}_\Gamma - \bar{\mathbf{x}}_\Gamma\|_2 + \|\mathbf{x}_{\Gamma^c} - \bar{\mathbf{x}}_{\Gamma^c}\|_2 = \|\mathbf{x}_\Gamma\|_2 + \|\mathbf{x}_{\Gamma^c} - \bar{\mathbf{x}}_{\Gamma^c}\|_2. \tag{5.46}$$

Then we have

$$\|\mathbf{x}_\Gamma - \bar{\mathbf{x}}_\Gamma\|_2 = \left\| \mathbf{x}_\Gamma - \Phi_\Gamma^\dagger \left(\Phi\mathbf{x}_\Gamma + \Phi\mathbf{x}_{\Gamma^c} + \mathbf{e} \right) \right\|_2 \tag{5.47a}$$

$$= \left\| \Phi_\Gamma^\dagger \left(\Phi\mathbf{x}_{\Gamma^c} + \mathbf{e} \right) \right\|_2 \tag{5.47b}$$

$$\leqslant \left\| \left(\Phi_\Gamma^\mathsf{T} \Phi_\Gamma \right)^{-1} \Phi_\Gamma^\mathsf{T} \Phi\mathbf{x}_{\Gamma^c} \right\|_2 + \left\| \Phi_\Gamma^\dagger \mathbf{e} \right\|_2 \tag{5.47c}$$

$$\leqslant \frac{1}{1 - \delta_{3s}} \left\| \Phi_\Gamma^\mathsf{T} \Phi\mathbf{x}_{\Gamma^c} \right\|_2 + \frac{1}{\sqrt{1 - \delta_{3s}}} \|\mathbf{e}\|_2 \tag{5.47d}$$

$$\leqslant \frac{\delta_{4s}}{1 - \delta_{3s}} \|\mathbf{x}_{\Gamma^c}\|_2 + \frac{1}{\sqrt{1 - \delta_{3s}}} \|\mathbf{e}\|_2, \tag{5.47e}$$

using Corollary 3.6 and Proposition 3.4. Then since $\delta_{3s} \leqslant \delta_{4s} \leqslant 0.1$ we get

$$\|\mathbf{x} - \bar{\mathbf{x}}\|_2 \leqslant \left(1 + \frac{\delta_{4s}}{1 - \delta_{3s}} \right) \|\mathbf{x}_{\Gamma^c}\|_2 + \frac{1}{\sqrt{1 - \delta_{3s}}} \|\mathbf{e}\|_2 \tag{5.48a}$$

$$\leqslant 1.112 \|\mathbf{x}_{\Gamma^c}\|_2 + 1.06 \|\mathbf{e}\|_2, \tag{5.48b}$$

proving the lemma. $\qquad\square$

Lemma 5.10 (Pruning, Lemma 4.5 of (Needell and Tropp 2008)). *The pruned approximation $\mathbf{x}^{(k)} = Prune(\bar{\mathbf{x}})$ where $\bar{\mathbf{x}}$ is the solution to the least squares problem $\arg\min_{\hat{\mathbf{x}}:\mathrm{supp}(\hat{\mathbf{x}})=\Gamma} \|\Phi\hat{\mathbf{x}} - y\|_2$ satisfies*

$$\left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2 \leqslant 2 \|\mathbf{x} - \bar{\mathbf{x}}\|_2. \tag{5.49}$$

Proof. We have

$$\left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2 \leqslant \|\mathbf{x} - \bar{\mathbf{x}}\|_2 + \left\| \mathbf{x}^{(k)} - \bar{\mathbf{x}} \right\|_2 \leqslant 2 \|\mathbf{x} - \bar{\mathbf{x}}\|_2, \tag{5.50}$$

as $\mathbf{x}^{(k)}$ is the best s-sparse approximation to $\bar{\mathbf{x}}$ and as $\mathbf{x}$ is also s-sparse, it must be further away from $\bar{\mathbf{x}}$ under the ℓ_2 norm. $\qquad\square$

Proof of Theorem 5.6. We now complete the proof of the error bounds of Theorem 5.6, recall that

this is for the case when $\mathbf{x}$ is s-sparse. Using the above lemmas we get

$$\left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2 = 2 \left\| \mathbf{x} - \bar{\mathbf{x}} \right\|_2 \qquad \text{(by Lemma 5.10)} \qquad (5.51a)$$

$$\leqslant 2 \left(1.112 \left\| \mathbf{x}_{\Gamma^c} \right\|_2 + 1.06 \left\| \mathbf{e} \right\|_2 \right) \qquad \text{(by Lemma 5.9)} \qquad (5.51b)$$

$$\leqslant 2.224 \left\| \mathbf{r}_{\Omega^c} \right\|_2 + 2.12 \left\| \mathbf{e} \right\|_2 \qquad \text{(by Lemma 5.8)} \qquad (5.51c)$$

$$\leqslant 2.224 \left(0.2223 \left\| \mathbf{r} \right\|_2 + 2.34 \left\| \mathbf{e} \right\|_2 \right) + 2.12 \left\| \mathbf{e} \right\|_2 \qquad \text{(by Lemma 5.7)} \qquad (5.51d)$$

$$\leqslant \frac{1}{2} \left\| \mathbf{r} \right\|_2 + \frac{15}{2} \left\| \mathbf{e} \right\|_2 \qquad (5.51e)$$

$$= \frac{1}{2} \left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2 + \frac{15}{2} \left\| \mathbf{e} \right\|_2. \qquad (5.51f)$$

The second bound of the theorem follows by solving this recursion (see Lemma 6.3). $\qquad \square$

The case where $\mathbf{x}$ in $\mathbf{y} = \Phi \mathbf{x} + \mathbf{e}$ is *not* sparse, as in Theorem 5.5, can be covered by assuming that we view the measurements $\mathbf{y}$ as coming from the sparse scenario, but with an additional error component. Recall Lemma 3.8 which states that if $\mathbf{y} = \Phi \mathbf{x} + \mathbf{e}$ then we can write $\mathbf{y} = \Phi \mathbf{x}^s + \tilde{\mathbf{e}}$ where

$$\left\| \tilde{\mathbf{e}} \right\|_2 \leqslant 1.05 \left(\left\| \mathbf{x} - \mathbf{x}^s \right\|_2 + \frac{1}{\sqrt{s}} \left\| \mathbf{x} - \mathbf{x}^s \right\|_1 \right) + \left\| \mathbf{e} \right\|_2. \qquad (5.52)$$

Proof of Theorem 5.5. Suppose we have the measurements $\mathbf{y} = \Phi \mathbf{x} + \mathbf{e}$ for $\Phi \in \mathbb{R}^{m \times n}$, $\mathbf{x}, \mathbf{e} \in \mathbb{R}^n$ and $\mathbf{y} \in \mathbb{R}^m$. Then we can use Lemma 3.8 to write $\mathbf{y} = \Phi \mathbf{x}^s + \tilde{\mathbf{e}}$ where $\mathbf{x}^s$ is s-sparse. Then from Theorem 5.6 we have

$$\left\| \mathbf{x}^s - \mathbf{x}^{(k+1)} \right\|_2 \leqslant \frac{1}{2} \left\| \mathbf{x}^s - \mathbf{x}^{(k)} \right\|_2 + \frac{15}{2} \left\| \tilde{\mathbf{e}} \right\|_2. \qquad (5.53)$$

Writing $\mathbf{x} - \mathbf{x}^{(k+1)} = \mathbf{x} - \mathbf{x}^s + \mathbf{x}^s - \mathbf{x}^{(k+1)}$ we obtain

$$\left\| \mathbf{x} - \mathbf{x}^{(k+1)} \right\|_2 \leqslant \left\| \mathbf{x} - \mathbf{x}^s \right\|_2 + \left\| \mathbf{x}^s - \mathbf{x}^{(k+1)} \right\|_2 \qquad (5.54a)$$

$$\leqslant \frac{3}{2} \left\| \mathbf{x} - \mathbf{x}^s \right\|_2 + \frac{1}{2} \left\| \mathbf{x}^s - \mathbf{x}^{(k)} \right\|_2 + \frac{15}{2} \left\| \tilde{\mathbf{e}} \right\|_2 \qquad (5.54b)$$

$$\leqslant \frac{1}{2} \left\| \mathbf{x}^s - \mathbf{x}^{(k)} \right\|_2 + 9.375 \left\| \mathbf{x} - \mathbf{x}^s \right\|_2 + \frac{7.875}{\sqrt{s}} \left\| \mathbf{x} - \mathbf{x}^s \right\|_1 + \frac{15}{2} \left\| \mathbf{e} \right\|_2 \quad \text{(by Lemma 3.8)}$$
$$\qquad (5.54c)$$

$$\leqslant \frac{1}{2} \left\| \mathbf{x}^s - \mathbf{x}^{(k)} \right\|_2 + 10\nu, \qquad (5.54d)$$

where ν is the unrecoverable energy as in Equation (5.31), namely $\nu = \left\| \mathbf{x} - \mathbf{x}^s \right\|_2 + \frac{1}{\sqrt{s}} \left\| \mathbf{x} - \mathbf{x}^s \right\|_1 + \left\| \mathbf{e} \right\|_2$. Again the second bound follows by solving the recursion. $\qquad \square$

Note that in Equation (5.54b) the $\frac{3}{2}$ term is taken from the original paper (Needell and Tropp 2008). There appears to be no reason why we this extra half term is needed, however the only effect is to lower the final bound to 9ν instead of 10ν.

The final step is to prove Theorem 5.4.

Proof of Theorem 5.4. Let η be some precision parameter, then setting $k \sim \mathcal{O}\left(\frac{\log \|\mathbf{x}\|_2}{\eta} \right)$ and using Theorem 5.5, CoSaMP produces a solution $\hat{\mathbf{x}}$ after k iterations that satisfies

$$\left\| \mathbf{x} - \hat{\mathbf{x}} \right\|_2 \leqslant C(\eta + \nu), \qquad (5.55)$$

where ν is the unrecoverable energy and C is a constant. Using Lemma 7 from (Gilbert, Strauss, Tropp, and Vershynin 2007) which states that

$$\nu \leqslant \frac{1.71}{\sqrt{s}} \left\| \mathbf{x} - \mathbf{x}^{s/2} \right\|_2 + \|\mathbf{e}\|_2 \tag{5.56}$$

we get the desired result, namely that

$$\|\mathbf{x} - \hat{\mathbf{x}}\|_2 \leqslant C \max \left\{ \eta, \frac{1}{\sqrt{s}} \left\| \mathbf{x} - \mathbf{x}^{s/2} \right\|_1 + \|\mathbf{e}\|_2 \right\}. \tag{5.57}$$

We will not show show the running time, but the result follows by maintaining a QR-decomposition of Φ_Γ when running the algorithm. $\qquad \square$

What is of concern however, is that the in proving the lemmas required for this theorem, the assumption was made that $\delta_{4s} \leqslant 0.1$, whereas the hypothesis of the theorem only requires that $\delta_{2s} \leqslant c$ for some constant c. It should be noted that the authors claim that this is covered by their Corollary 3.4 of (Needell and Tropp 2008).

5.2 Iterative Thresholding Algorithms

Thresholding algorithms are a class of algorithms that perform some thresholding function on each iteration. A typical iteration would like something like

$$\mathbf{x}^{(i)} = \mathbb{T}_\tau \left(f(\mathbf{x}^{(i-1)}) \right), \tag{5.58}$$

where $\mathbb{T}_\tau$ is a thresholding function with parameters τ, and f is some function that acts on the output of the previous iterate. We distinguish between *hard thresholding* and *soft thresholding*. In the former case we will often write $\mathbb{H}_s$ for the thresholding function where s is the number of components that are non-zero and for soft thresholding we will often use $\mathbb{S}_\tau$.

In section 5.2.1 we will demonstrate an iterative hard thresholding algorithm and in section 5.2.2 we will show an iterative soft thresholding algorithm. In Chapter 6 we will present another iterative hard thresholding algorithm, although similar to the other hard thresholding algorithm, it has some significantly different properties.

5.2.1 Hard Thresholding

Hard thresholding makes use of a thresholding function $\mathbb{H}_s : \mathbb{R}^n \to \mathbb{R}^n$ that sets all but the s-largest component (in magnitude) of a vector $\mathbf{z} \in \mathbb{R}^n$ to 0 and leaves the remaining components untouched, that is, if ξ is the s-largest component of $\mathbf{x}$, then

$$\mathbb{H}_s(\mathbf{x}) = \begin{cases} x_i & \text{if } |x_i| \geqslant |\xi| \\ 0 & \text{otherwise.} \end{cases} \tag{5.59}$$

We may regard $\mathbb{H}_s(\mathbf{z})$ as the best s-sparse approximation to $\mathbf{z}$, that is, $\|\mathbf{z} - \mathbb{H}_s(\mathbf{z})\|_1$ is minimal, which was shown in Lemma 2.14. If there is no unique such vector, i.e. there is no unique s-largest component, the tie can be broken deterministically (e.g. by lexicographical ordering) or randomly.

The full algorithm can be seen in Algorithm 5.5 but the critical update step is

$$\mathbf{x}^{(i)} = \mathbb{H}_s\left(\mathbf{x}^{(i-1)} + \Phi^\mathsf{T}\left(\mathbf{y} - \Phi\mathbf{x}^{(i-1)}\right)\right). \tag{5.60}$$

This algorithm is very similar to the frame reconstruction algorithm, Algorithm 4.1, but we will discuss some of the differences in Chapter 6.

Algorithm 5.5 Iterative Hard Thresholding

Input:

- ▶ s the sparsity of $\mathbf{x}$
- ▶ $\mathbf{y} \in \mathbb{R}^m$ and the measurement matrix $\Phi \in \mathbb{R}^{m \times n}$

Output:

- ▶ $\hat{\mathbf{x}} \in \mathbb{R}^n$ such that $\hat{\mathbf{x}}$ is s sparse and $\mathbf{y} = \Phi\hat{\mathbf{x}}$

1: $\mathbf{x}^{(0)} \leftarrow \mathbf{0}$
2: **for** $i = 1, \ldots$ **do**
3:     $\mathbf{x}^{(i)} \leftarrow \mathbb{H}_s\left(\mathbf{x}^{(i-1)} + \Phi^\mathsf{T}\left(\mathbf{y} - \Phi\mathbf{x}^{(i-1)}\right)\right)$
4: **end for**
5: $\hat{\mathbf{x}} \leftarrow \mathbf{x}^{(i)}$
6: **return** $\hat{\mathbf{x}}$

The main theorem of (Blumensath and Davies 2008, Theorem 1), quantifies the performance of Algorithm 5.5. The analysis of this algorithm requires the modified restricted isometry property (modified-RIP), defined earlier.

Theorem 5.11. *Given a noisy observation* $\mathbf{y} = \Phi\mathbf{x} + \mathbf{e}$ *where* $\mathbf{x}, \mathbf{e} \in \mathbb{R}^n$, $\mathbf{y} \in \mathbb{R}^m$ *and* $\Phi \in \mathbb{R}^{m \times n}$, *then let* $\mathbf{x}^s \in \mathbb{R}^n$ *be an s-sparse approximation to* $\mathbf{x}$ *such that* $\|\mathbf{x} - \mathbf{x}^s\|_2$ *is minimal. If* Φ *obeys the modified-RIP with* $\beta_{3s} < 1/\sqrt{32}$, *then at iteration k, the iterative hard thresholding algorithm will recover an approximation* $\mathbf{x}^{(k)}$ *satisfying*

$$\left\|\mathbf{x} - \mathbf{x}^{(k)}\right\|_2 \leqslant 2^{-k}\|\mathbf{x}^s\|_2 + 5\tilde{\varepsilon}_s, \tag{5.61}$$

where

$$\tilde{\varepsilon}_s = \|\mathbf{x} - \mathbf{x}^s\|_2 + \frac{1}{\sqrt{s}}\|\mathbf{x} - \mathbf{x}^s\|_1 + \|\mathbf{e}\|_2, \tag{5.62}$$

is the unrecoverable energy.

Furthermore after at most

$$k^\star = \left\lceil \log_2\left(\frac{\|\mathbf{x}^s\|_2}{\tilde{\varepsilon}_s}\right)\right\rceil, \tag{5.63}$$

iterations the algorithm estimates $\mathbf{x}$ *with accuracy*

$$\left\|\mathbf{x} - \mathbf{x}^{(k^\star)}\right\|_2 \leqslant 6\left(\|\mathbf{x} - \mathbf{x}^s\|_2 + \frac{1}{\sqrt{s}}\|\mathbf{x} - \mathbf{x}^s\|_1 + \|\mathbf{e}\|_2\right) = 6\tilde{\varepsilon}_s. \tag{5.64}$$

For signals $\mathbf{x}$ that are s-sparse we have the following corollary.

Corollary 5.12. *Given a noisy observation* $\mathbf{y} = \Phi\mathbf{x} + \mathbf{e}$ *where* $\mathbf{x} \in \mathbb{R}^n$ *is s-sparse and if* Φ *has the modified-RIP with* $\beta_{3s} < 1/\sqrt{32}$, *then, at iteration k, Algorithm 5.5 will recover an approximation* $\mathbf{x}^{(k)}$ *satisfying*

$$\left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2 \leqslant 2^{-k} \|\mathbf{x}\|_2 + 4 \|\mathbf{e}\|_2. \tag{5.65}$$

Furthermore after at most

$$k^{\star} = \left\lceil \log_2 \left(\frac{\|\mathbf{x}^s\|_2}{\|\mathbf{e}\|_2} \right) \right\rceil, \tag{5.66}$$

iterations the algorithm estimates $\mathbf{x}$ *with accuracy*

$$\left\| \mathbf{x} - \mathbf{x}^{(k^{\star})} \right\|_2 \leqslant 5 \|\mathbf{e}\|_2 \tag{5.67}$$

We will now prove Theorem 5.11 and Corollary 5.12, following the proof of (Blumensath and Davies 2008). Let us recall and make the following definitions:

$$\mathbf{r}^{(i)} \triangleq \mathbf{x}^s - \mathbf{x}^{(i)}, \tag{5.68a}$$

$$\mathbf{a}^{(i)} \triangleq \mathbf{x}^{(i-1)} + \Phi^{\mathsf{T}}(\mathbf{y} - \Phi\mathbf{x}^{(i-1)}), \tag{5.68b}$$

$$\mathbf{x}^{(i)} \triangleq \mathbb{H}_s(\mathbf{a}^{(i)}), \tag{5.68c}$$

$$\Gamma^{\star} \triangleq \mathrm{supp}(\mathbf{x}^s), \tag{5.68d}$$

$$\Gamma^{(i)} \triangleq \mathrm{supp}(\mathbf{x}^{(i)}), \tag{5.68e}$$

$$B^{(i)} \triangleq \Gamma^{\star} \cup \Gamma^{(i)}, \tag{5.68f}$$

$$\hat{\mathbf{x}} = \mathbf{x}_\Gamma \triangleq \{\hat{x}_i = x_i \text{ if } i \in \Gamma, \text{ else } \hat{x}_i = 0\}. \tag{5.68g}$$

As a consequence we have $|\Gamma^{\star}| \leqslant s$ and $\left|\Gamma^{(i)}\right| \leqslant s$.

Proof. Consider the error $\left\| \mathbf{x}^s - \mathbf{x}^{(i+1)} \right\|_2$. Applying the triangle inequality to this we get

$$\left\| \mathbf{x}^s - \mathbf{x}^{(i+1)} \right\|_2 \leqslant \left\| \mathbf{x}^s_{B^{(i+1)}} - \mathbf{a}^{(i+1)}_{B^{(i+1)}} \right\|_2 + \left\| \mathbf{x}^{(i+1)}_{B^{(i+1)}} - \mathbf{a}^{(i+1)}_{B^{(i+1)}} \right\|_2 \tag{5.69}$$

as $\mathbf{x}^s - \mathbf{x}^{(i+1)}$ is supported on the set $B^{(i+1)}$. As $\mathbf{x}^{(i+1)}$ is the thresholded version of $\mathbf{a}^{(i+1)}$ it is the best s-term approximation to $\mathbf{a}^{(i+1)}$, in particular it is better than $\mathbf{x}^s$. Hence

$$\left\| \mathbf{x}^{(i+1)} - \mathbf{a}^{(i+1)}_{B^{(i+1)}} \right\|_2 \leqslant \left\| \mathbf{x}^s - \mathbf{a}^{(i+1)}_{B^{(i+1)}} \right\|_2, \tag{5.70}$$

and thus Equation (5.69) becomes

$$\left\| \mathbf{x}^s - \mathbf{x}^{(i+1)} \right\|_2 \leqslant 2 \left\| \mathbf{x}^s_{B^{(i+1)}} - \mathbf{a}^{(i+1)}_{B^{(i+1)}} \right\|_2. \tag{5.71}$$

Using the fact that $\mathbf{y} = \Phi\mathbf{x}^s + \mathbf{e}$ and $\mathbf{r}^{(i)} = \mathbf{x}^s - \mathbf{x}^{(i)}$ we get

$$\mathbf{a}^{(i+1)}_{B^{(i+1)}} = \mathbf{x}^{(i)}_{B^{(i+1)}} + \Phi^{\mathsf{T}}_{B^{(i+1)}} \left(\mathbf{y} - \Phi\mathbf{x}^{(i)}_{B^{(i+1)}} \right) \tag{5.72a}$$

$$= \mathbf{x}^{(i)}_{B^{(i+1)}} + \Phi^{\mathsf{T}}_{B^{(i+1)}} \Phi\mathbf{r}^{(i)} + \Phi^{\mathsf{T}}_{B^{(i+1)}} \mathbf{e}, \tag{5.72b}$$

hence

$$\left\| \mathbf{x}^s - \mathbf{x}^{(i+1)} \right\|_2 \tag{5.73a}$$

$$\leqslant 2 \left\| \underbrace{\mathbf{x}^s_{B^{(i+1)}} - \mathbf{x}^{(i)}_{B^{(i+1)}}}_{\mathbf{r}^{(i)}_{B^{(i+1)}}} - \Phi^{\mathsf{T}}_{B^{(i+1)}} \Phi \mathbf{r}^{(i)} - \Phi^{\mathsf{T}}_{B^{(i+1)}} \mathbf{e} \right\|_2 \tag{5.73b}$$

$$\leqslant 2 \left\| \mathbf{r}^{(i)}_{B^{(i+1)}} - \Phi^{\mathsf{T}}_{B^{(i+1)}} \Phi \mathbf{r}^{(i)} \right\|_2 + 2 \left\| \Phi^{\mathsf{T}}_{B^{(i+1)}} \mathbf{e} \right\|_2 \tag{5.73c}$$

$$= 2 \left\| \left(\mathbf{I} - \Phi^{\mathsf{T}}_{B^{(i+1)}} \Phi_{B^{(i+1)}} \right) \mathbf{r}^{(i)}_{B^{(i+1)}} - \Phi^{\mathsf{T}}_{B^{(i+1)}} \Phi_{B^{(i)} \backslash B^{(i+1)}} \mathbf{r}^{(i)}_{B^{(i)} \backslash B^{(i+1)}} \right\|_2 + 2 \left\| \Phi^{\mathsf{T}}_{B^{(i+1)}} \mathbf{e} \right\|_2 \tag{5.73d}$$

$$\leqslant 2 \left\| \left(\mathbf{I} - \Phi^{\mathsf{T}}_{B^{(i+1)}} \Phi_{B^{(i+1)}} \right) \mathbf{r}^{(i)}_{B^{(i+1)}} \right\|_2 + 2 \left\| \Phi^{\mathsf{T}}_{B^{(i+1)}} \Phi_{B^{(i)} \backslash B^{(i+1)}} \mathbf{r}^{(i)}_{B^{(i)} \backslash B^{(i+1)}} \right\|_2 + 2 \left\| \Phi^{\mathsf{T}}_{B^{(i+1)}} \mathbf{e} \right\|_2, \tag{5.73e}$$

by repeated application of the triangle inequality. Then

$$\left| B^{(i)} \cup B^{(i+1)} \right| = \left| \Gamma^\star \cup \Gamma^{(i)} \cup \Gamma^{(i+1)} \right| \leqslant 3s, \tag{5.74}$$

as each set $\Gamma^{(\cdot)}$ has only s entries. Recall from the modified RIP and Lemmas 3.10 and 3.11 we have

$$\left\| \Phi^{\mathsf{T}}_\Lambda \mathbf{x} \right\|_2 \leqslant \|\mathbf{x}\|_2, \tag{5.75a}$$

$$\left\| \left(\mathbf{I} - \Phi^{\mathsf{T}}_\Lambda \Phi_\Lambda \right) \mathbf{x}_\Lambda \right\|_2 \leqslant \beta_s \|\mathbf{x}_\Lambda\|_2, \tag{5.75b}$$

$$\left\| \Phi^{\mathsf{T}}_\Lambda \Phi_{\Lambda'} \mathbf{x}_{\Lambda'} \right\|_2 \leqslant \beta_s \|\mathbf{x}_\Lambda\|_2, \tag{5.75c}$$

for all matrices Φ which obey the RIP and disjoint sets Λ, Λ' of size s. We also have $\beta_{2s} \leqslant \beta_{3s}$. Therefore we can write Equation (5.73e) as

$$\left\| \mathbf{r}^{(i+1)} \right\|_2 \tag{5.76a}$$

$$\leqslant 2 \underbrace{\left\| \left(\mathbf{I} - \Phi^{\mathsf{T}}_{B^{(i+1)}} \Phi_{B^{(i+1)}} \right) \mathbf{r}^{(i)}_{B^{(i+1)}} \right\|_2}_{(5.75b)} + 2 \underbrace{\left\| \Phi^{\mathsf{T}}_{B^{(i+1)}} \Phi_{B^{(i)} \backslash B^{(i+1)}} \mathbf{r}^{(i)}_{B^{(i)} \backslash B^{(i+1)}} \right\|_2}_{(5.75c)} + 2 \underbrace{\left\| \Phi^{\mathsf{T}}_{B^{(i+1)}} \mathbf{e} \right\|_2}_{(5.75a)}, \tag{5.76b}$$

$$\leqslant 2\beta_{2s} \left\| \mathbf{r}^{(i)}_{B^{(i+1)}} \right\|_2 + 2\beta_{3s} \left\| \mathbf{r}^{(i)}_{B^{(i)} \backslash B^{(i+1)}} \right\|_2 + 2 \|\mathbf{e}\|_2 \tag{5.76c}$$

$$\leqslant 2\beta_{3s} \left(\left\| \mathbf{r}^{(i)}_{B^{(i+1)}} \right\|_2 + \left\| \mathbf{r}^{(i)}_{B^{(i)} \backslash B^{(i+1)}} \right\|_2 \right) + 2 \|\mathbf{e}\|_2 \tag{5.76d}$$

$$\leqslant 2\sqrt{2}\beta_{3s} \left\| \mathbf{r}^{(i)} \right\|_2 + 2 \|\mathbf{e}\|_2, \tag{5.76e}$$

where the last step follows since $\mathbf{r}^{(i)}_{B^{(i+1)}}$ and $\mathbf{r}^{(i)}_{B^{(i)} \backslash B^{(i+1)}}$ are orthogonal and for any two orthogonal vectors $\mathbf{u}$ and $\mathbf{v}$ we have

$$\|\mathbf{u}\|_2 + \|\mathbf{v}\|_2 \leqslant \sqrt{2} \|\mathbf{u} + \mathbf{v}\|_2. \tag{5.77}$$

If $\beta_{3s} < \frac{1}{\sqrt{32}}$ we get

$$\left\| \mathbf{r}^{(i+1)} \right\|_2 \leqslant \frac{\left\| \mathbf{r}^{(i)} \right\|_2}{2} + 2 \|\mathbf{e}\|_2. \tag{5.78}$$

Expanding this using Lemma 6.3 and using the fact that we start with $\mathbf{x}^{(0)} = \mathbf{0}$ we get on the k-th iteration

$$\left\| \mathbf{r}^{(k)} \right\|_2 < 2^{-k} \|\mathbf{x}^s\|_2 + 4 \|\mathbf{e}\|_2, \tag{5.79}$$

proving Corollary 5.12.

To prove the main theorem we require a lemma from (Needell and Tropp 2008), Lemma 3.8. We repeat the statements from these lemmas here for convenience. For a matrix $\Phi \in \mathbb{R}^{m \times n}$ that satisfies the RIP of order s and if $\mathbf{x}^s$ is the best s-term approximation to $\mathbf{x}$ where $\mathbf{y} = \Phi\mathbf{x} + \mathbf{e}$, then we have the following bound on the error $\tilde{\mathbf{e}}$ where $\mathbf{y} = \Phi\mathbf{x}^s + \tilde{\mathbf{e}}$,

$$\|\tilde{\mathbf{e}}\|_2 \leqslant \|\mathbf{x} - \mathbf{x}^s\|_2 + \frac{1}{\sqrt{2}} \|\mathbf{x} - \mathbf{x}^s\|_1 + \|\mathbf{e}\|_2 . \tag{5.80}$$

To bound the error $\left\|\mathbf{x} - \mathbf{x}^{(i)}\right\|_2$ we use the triangle inequality and add some extra terms

$$\left\|\mathbf{x} - \mathbf{x}^{(i)}\right\|_2 \leqslant \left\|\mathbf{r}^{(i)}\right\|_2 + \|\mathbf{x} - \mathbf{x}^s\|_2 \tag{5.81a}$$

$$\leqslant \left\|\mathbf{r}^{(i)}\right\|_2 + \|\mathbf{x} - \mathbf{x}^s\|_2 + \frac{1}{\sqrt{2}} \|\mathbf{x} - \mathbf{x}^s\|_1 + \|\mathbf{e}\|_2 \tag{5.81b}$$

$$= \left\|\mathbf{r}^{(i)}\right\|_2 + \tilde{\varepsilon}_s, \tag{5.81c}$$

where $\tilde{\varepsilon}_s = \|\mathbf{x} - \mathbf{x}^s\|_2 + \frac{1}{\sqrt{2}} \|\mathbf{x} - \mathbf{x}^s\|_1 + \|\mathbf{e}\|_2$. From Corollary 5.12 we have

$$\left\|\mathbf{r}^{(i)}\right\|_2 \leqslant 2^{-i} \|\mathbf{x}^s\|_2 + 4 \|\tilde{\mathbf{e}}\|_2 . \tag{5.82}$$

From Lemma 3.8 we have that

$$\|\tilde{\mathbf{e}}\|_2 \leqslant \|\mathbf{x} - \mathbf{x}^s\|_2 + \frac{1}{\sqrt{2}} \|\mathbf{x} - \mathbf{x}^s\|_1 + \|\mathbf{e}\|_2 = \tilde{\varepsilon}_s \tag{5.83}$$

so

$$\left\|\mathbf{r}^{(i)}\right\|_2 \leqslant 2^{-i} \|\mathbf{x}^s\|_2 + 4\tilde{\varepsilon}_s, \tag{5.84}$$

and hence

$$\left\|\mathbf{x} - \mathbf{x}^{(i)}\right\|_2 \leqslant 2^{-i} \|\mathbf{x}^s\|_2 + 5\tilde{\varepsilon}_s. \tag{5.85}$$

It remains only to show the bound on the iteration count. To recover $\mathbf{x}$ with an error of less than $6\tilde{\varepsilon}_s$ we require that

$$2^{-i} \|\mathbf{x}^s\|_2 \leqslant \tilde{\varepsilon}_s \quad \Longleftrightarrow \quad 2^k \geqslant \frac{\|\mathbf{x}^s\|_2}{\tilde{\varepsilon}_s}, \tag{5.86}$$

which implies that

$$k \geqslant \log_2 \left(\frac{\|\mathbf{x}^s\|_2}{\tilde{\varepsilon}_s} \right), \tag{5.87}$$

and the second part of the theorem follows immediately. $\qquad\square$

5.2.2 Soft Thresholding

Soft thresholding algorithms bear great similarity to hard thresholding algorithms. Soft thresholding can be used to minimise equations of the form

$$F_\tau(\mathbf{x}) \triangleq \|\Phi\mathbf{x} - \mathbf{y}\|_2^2 + 2\tau \|\mathbf{x}\|_1 . \tag{5.88}$$

It was shown in (Daubechies, Defrise, and Mol 2004) that the solution to this equation is given by the limit of the sequence

$$\mathbf{x}^{(k+1)} = S_\tau \left(\mathbf{x}^{(k)} + \Phi^\mathsf{T} \mathbf{y} - \Phi^\mathsf{T} \Phi \mathbf{x}^{(k)} \right), \tag{5.89}$$

where $S_\tau(\mathbf{x})_j = S_\tau(x_j)$ is the soft thresholding function

$$S_\tau(x) \triangleq \begin{cases} x - \tau & \text{if } x > \tau \\ 0 & \text{if } |x| \leqslant \tau \\ x + \tau & \text{if } x < -\tau, \end{cases} \tag{5.90}$$

applied component-wise to each element of the input vector.

The iterative step in Equation (5.89) is known as the *Landweber iteration*. More recent work in (Daubechies, Fornasier, and Loris 2007) suggests a slightly different form of this equation, which they call the *projected Landweber iteration*. Let us make the following definitions first. Define $\mathcal{B}_R \subset \mathbb{R}^n$ to be the ℓ_1 ball

$$\mathcal{B}_R \triangleq \{ \mathbf{x} \in \mathbb{R}^n \colon \|\mathbf{x}\|_1 \leqslant R \}, \tag{5.91}$$

of radius R. Then let $\mathbb{P}_C : \mathbb{R}^n \to \mathbb{R}^n$ be the projection of a point $\mathbf{x} \in \mathbb{R}^n$ to the closest point (under the ℓ_2 norm) onto the convex set C. As in (Daubechies, Fornasier, and Loris 2007) we will abuse notation and write $\mathbb{P}_R$ when we mean $\mathbb{P}_{\mathcal{B}_R}$. Then the projected Landweber iteration is given by

$$\mathbf{x}^{(k+1)} = \mathbb{P}_R \left(\mathbf{x}^{(k)} + \Phi^\mathsf{T} \mathbf{y} - \Phi^\mathsf{T} \Phi \mathbf{x}^{(k)} \right). \tag{5.92}$$

The following lemma from (Daubechies, Fornasier, and Loris 2007) shows how this can be efficiently implemented.

Lemma 5.13 (Lemma 2 from (Daubechies, Fornasier, and Loris 2007)). *Let $\mathbf{x} \in \mathbb{R}^n$ and let $R \in \mathbb{R}$ be a positive constant. If $\|\mathbf{x}\|_1 > R$, then the ℓ_2 projection of $\mathbf{x}$ on the ℓ_1 ball with radius R is given by $\mathbb{P}_R(\mathbf{x}) = S_\mu(\mathbf{x})$ where μ (depending on $\mathbf{x}$ and R) is chosen such that $\|S_\mu(\mathbf{x})\|_1 = R$. If $\|\mathbf{x}\|_1 \leqslant R$ then $\mathbb{P}_R(\mathbf{x}) = S_0(\mathbf{x}) = \mathbf{x}$.*

In Algorithm 5.6 we give an implementation to calculate $\mathbb{P}_R(\mathbf{x})$ for arbitrary $\mathbf{x}$ based on this lemma. Using the terms from the algorithm, the output of the algorithm $\mathbf{y}$ has the following property

$$\|\mathbf{y}\|_1 = \|S_\mu(\mathbf{x})\|_1 \tag{5.93a}$$

$$= \sum_{i=1}^{n} \max\{|\mathbf{x}_i| - \mu, 0\} \tag{5.93b}$$

$$= \sum_{i=1}^{k} (\hat{\mathbf{x}}_i - \mu) \tag{5.93c}$$

$$= \sum_{i=1}^{k-1} (\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_k) \tag{5.93d}$$

$$= \|S_{\hat{\mathbf{x}}_k}(\mathbf{x})\|_1 + k\nu \tag{5.93e}$$

$$= R. \tag{5.93f}$$

Algorithm 5.6 ℓ_2 projection onto ℓ_1 ball

Input:

- ▶ A vector $\mathbf{x} \in \mathbb{R}^n$.
- ▶ A radius $R > 0$.

Output:

- ▶ A vector $\mathbf{y} \in \mathbb{R}^n$ so that $\mathbf{y}$ is the closest point (under Euclidean or ℓ_2 distance) in the ℓ_1 ball of radius R to $\mathbf{x}$.

1: **if** $\|\mathbf{x}\|_1 \leqslant R$ **then**
2:     **return** $\mathbf{y} \leftarrow \mathbf{x}$
3: **end if**
4: Sort the components of $\mathbf{x}$ by magnitude to get the vector $\hat{\mathbf{x}}$ where $|\hat{\mathbf{x}}_1| \geqslant |\hat{\mathbf{x}}_2| \geqslant \cdots \geqslant |\hat{\mathbf{x}}_n|$.
5: Find k such that

$$\left\| \mathsf{S}_{\hat{\mathbf{x}}_k}(\mathbf{x}) \right\|_1 = \sum_{i=1}^{k-1} (\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_k) \leqslant R < \sum_{i=1}^{k} (\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_{k+1}) = \left\| \mathsf{S}_{\hat{\mathbf{x}}_{k+1}}(\mathbf{x}) \right\|_1 . \tag{5.94}$$

6: $\nu \leftarrow \frac{1}{k} \left(R - \left\| \mathsf{S}_{\hat{\mathbf{x}}_k}(\mathbf{x}) \right\|_1 \right)$
7: $\mu \leftarrow \hat{\mathbf{x}}_k + \nu$
8: $\mathbf{y} \leftarrow \mathsf{S}_\mu(\mathbf{x})$
9: **return** $\mathbf{y}$

One can also include an adaptive descent parameter $\beta^{(k)} > 0$ in each iteration, giving the iterative step

$$\mathbf{x}^{(k+1)} = \mathbb{P}_R \left(\mathbf{x}^{(k)} + \beta^{(k)} \Phi^{\mathsf{T}} \left(\mathbf{y} - \Phi \mathbf{x}^{(k)} \right) \right) . \tag{5.95}$$

The main theorem of (Daubechies, Fornasier, and Loris 2007) says that the projected Landweber iterative step with an adaptive descent parameter will converge to the minimiser $\hat{\mathbf{x}}$ in the ℓ_1 ball $\mathcal{B}_R$ of

$$\mathcal{D}(\mathbf{x}) \triangleq \|A\mathbf{x} - \mathbf{y}\|_2 , \tag{5.96}$$

for all matrices $A \in \mathbb{R}^{m \times n}$ and $y \in \mathbb{R}^m$, that is, it solves the problem

$$\arg\min_{\hat{\mathbf{x}} \in \mathcal{B}_R} \|A\mathbf{x} - \mathbf{y}\|_2 . \tag{5.97}$$

Theorem 5.14 (Theorem 1 of (Daubechies, Fornasier, and Loris 2007)). *The sequence* $\left\{ \mathbf{x}^{(k)} \right\}_{k \in \mathbb{N}}$ *given by*

$$\mathbf{x}^{(k+1)} = \mathbb{P}_R \left[\mathbf{x}^{(k)} + \beta^{(k)} \Phi^{\mathsf{T}} \left(\mathbf{y} - \Phi \mathbf{x}^{(k)} \right) \right] , \tag{5.98}$$

where the step length $\beta^{(k)}$ *satisfies*

(a) $\bar{\beta} \triangleq \sup \left\{ \beta^{(n)} \colon \in \mathbb{N} \right\} < \infty$ *and* $\inf \left\{ \beta^{(n)} \colon \in \mathbb{N} \right\} \geqslant 1$, *and*

(b) *if there exists* $n_0 \in \mathbb{N}$ *such that*

$$\beta^{(n)} \left\| \Phi \left(\mathbf{x}^{(n+1)} - \mathbf{x}^{(n)} \right) \right\|_2^2 \leqslant r \left\| \mathbf{x}^{(n+1)} - \mathbf{x}^{(n)} \right\|_2 \qquad \forall n \geqslant n_0, \tag{5.99}$$

where the constant r *is given by* $r \triangleq \|\Phi^* \Phi\|_{\ell_2 \to \ell_2} < 1$,

converges in norm to a minimiser of $\|\Phi \mathbf{x} - \mathbf{y}\|_2$ *on* $\mathcal{B}_R$.

Algorithm 5.7 Soft Thresholding

Input:

▶ A vector $\mathbf{y} \in \mathbb{R}^m$ of observations.
▶ A matrix $\Phi \in \mathbb{R}^{m \times n}$ with eigenvalues less than 1.
▶ A radius R.

Output:

▶ A vector $\mathbf{x}_R \in \mathbb{R}^n$ that is the minimiser in B_R of $\mathcal{D}(\mathbf{x}) = \|\Phi\mathbf{x} - \mathbf{y}\|_2$.

1: $\mathbf{x}^{(0)} \leftarrow \mathbf{0}$
2: **for** $k = 1, \ldots, \infty$ **do**
3:     Select $\beta^{(k)}$ satisfying the conditions of Theorem 5.14.
4:         $\mathbf{x}^{(k)} \leftarrow \mathbb{P}_R \left[\mathbf{x}^{(k-1)} + \beta^{(k)} \Phi^\mathsf{T} \left(\mathbf{y} - \Phi\mathbf{x}^{(k-1)} \right) \right]$ ▷ Implemented via soft-thresholding
5: **end for**
6: $\mathbf{x}_R \leftarrow \mathbf{x}^{(k)}$

Note that the second condition on the $\beta^{(k)}$ requires that all the eigenvalues of Φ are strictly less than 1.

We can use the result of Theorem 5.14 to form an algorithm to find the sparsest solution of $\mathbf{y} = \Phi\mathbf{x} + \mathbf{e}$, which we list as Algorithm 5.7. Ideally we would set the radius equal to $\|\mathbf{x}\|_1$, but this is unknown. In practice one runs the algorithm for several values of R generating a set of vectors $\mathbf{x}_R$.

There are two simple ways of choosing the values of $\beta^{(k)}$:

▶ Set $\beta^{(k)} = 1$ for all k which trivially satisfies the conditions of the theorem.
▶ Use a greedy strategy to select $\beta^{(k)}$, for example

$$\beta^{(k)} = \frac{\left\| \Phi^\mathsf{T} \left(\mathbf{y} - \Phi\mathbf{x}^{(k)} \right) \right\|_2^2}{\left\| \Phi\Phi^\mathsf{T} \left(\mathbf{y} - \Phi\mathbf{x}^{(k)} \right) \right\|_2^2}. \tag{5.100}$$

More details can be found in (Figueiredo, Nowak, and Wright 2007).

Because of the condition on the eigenvalues of Φ, this does not work for the same matrices that we typically use for compressive sensing.

5.3   Model Based

Model based compressive sensing is a way of adding extra rules to the reconstruction process typically used for compressive sensing. The principle is that the reconstruction process can be improved by imposing extra guidelines, in particular, in assuming that the sparse coefficients have some extra underlying structure. What we have discussed so far in compressive sensing assumes nothing about the positions of the non-zero or large coefficients of the signal, but in many real world examples there is a strong structure present in the data which we can take advantage of to improve the reconstruction.

For instance when performing wavelet based compression on piecewise smooth signals, the large wavelet coefficients form a connected tree structure (Crouse, Nowak, and Baraniuk 1998). It is also not uncommon for the signals to exhibit *block sparsity* such as in DNA microarrays (Stojnic, Parvaresh, and Hassibi 2008), magnetoencephalography (Eldar and Mishali 2008), sensory networks (Baron, Duarte, Sarvotham, Wakin, and Baraniuk 2006) or MIMO communications (Wakin, Sarvotham, Duarte, Baron, and Baraniuk 2005).

Although there has been some work in taking advantage of the structure of the large coefficients, such as in (Blumensath and Davies 2007; Lu and Do 2008) this work has been only on looking at signals coming from the unions of particular subspaces and does not offer feasible recovery algorithms. The main claim of the recent work (Baraniuk, Cevher, Duarte, and Hegde 2008) is that it offers a general framework for model based recovery, *independent* of the model used, as well as generalising the RIP to a model based version, which they call the *restricted amplification property* (RAmP). Certainly the idea of using model information to improve such algorithms is not new, having been proposed for algorithms such as Bootstrap, leading to model-based bootstrap (Efron 1979; Efron 1981).

The work in (Baraniuk, Cevher, Duarte, and Hegde 2008) offers a way of incorporating model based compressive sensing into two existing algorithms, CoSaMP (Needell and Tropp 2008) and iterative hard thresholding (Blumensath and Davies 2008). Both of these algorithms feature a pruning or thresholding step in which the current best solution is forced to conform to the sparsity constraints. The idea of model based CS is to perform this prune using information from the model, rather than simply the positions of the largest coefficients.

To make sense of the theorems in (Baraniuk, Cevher, Duarte, and Hegde 2008) we need the following definitions. Unless otherwise stated the following theorems and definitions are from (Baraniuk, Cevher, Duarte, and Hegde 2008).

Definition 5.15. *A signal model $\mathcal{M}_K$ is defined as the union of m_K canonical K-dimensional subspaces*

$$\mathcal{M}_K \triangleq \bigcup_{m=1}^{m_K} \mathcal{X}_m \qquad such\ that\ \mathcal{X}_m \triangleq \{\mathbf{x} \in \mathbb{R}^n \colon \mathrm{supp}(\mathbf{x}) = \Omega_m\}. \tag{5.101}$$

A signal model $\mathcal{M}_K$ can be characterised by the set of supports $\{\Omega_m\}$ for $m = 1, \ldots, m_K$ and contains all the signals with these supports.

Definition 5.16. *The B-Minkowski sum $\mathcal{M}_K^N$ for the set $\mathcal{M}_K$, $B > 1$ is defined as*

$$\mathcal{M}_K^B \triangleq \left\{ \mathbf{x} = \sum_{b=1}^{B} \mathbf{x}^{(b)} \colon \mathbf{x}^{(b)} \in \mathcal{M}_K \right\}. \tag{5.102}$$

Then $\mathbb{M}_B(\mathbf{x}, K)$ is an algorithm that finds the best approximation to $\mathbf{x}$ contained in the union of subspaces $\mathcal{M}_K^B$, i.e.

$$\mathbb{M}_B(\mathbf{x}, K) \triangleq \arg \min_{\hat{\mathbf{x}} \in \mathcal{M}_K^B} \|\mathbf{x} - \hat{\mathbf{x}}\|_2. \tag{5.103}$$

We will write $\mathbb{M}(\mathbf{x}, K)$ when we mean $\mathbb{M}_1(\mathbf{x}, K)$.

Note that generally we have $\mathcal{M}_K^B \subset \mathcal{M}_{BK}$, hence $\mathbb{M}(\mathbf{x}, BK)$ will often produce a better approximation than $\mathbb{M}_B(\mathbf{x}, K)$.

Rather than use the RIP as before, we are now interested in matrices Φ that are near length

preserving for all vectors in $\mathcal{M}_K^B$, rather than all K-sparse vectors, we call this the $\mathcal{M}_K$-RIP. The following theorem from (Blumensath and Davies 2007) quantifies the probability that a matrix Φ generated from a subgaussian distribution obeys this new $\mathcal{M}_K$-RIP.

Theorem 5.17 ((Blumensath and Davies 2007)). *Let $\mathcal{M}_K$ be the union of m_K subspaces of dimension K in $\mathbb{R}^n$. Then for any $t > 0$ and any*

$$m \geqslant \frac{2}{c\delta_{\mathcal{M}_K}^s} \left(\log(2m_K) + K \log \frac{12}{\delta_{\mathcal{M}_K}} + t \right), \tag{5.104}$$

any matrix $\Phi \in \mathbb{R}^{m \times n}$ drawn from a subgaussian distribution obeys the $\mathcal{M}_K$-RIP with constant $\delta_{\mathcal{M}_K}$-RIP with probability at least $1 - e^{-t}$.

Similar to before, we define $\sigma_{\mathcal{M}_K}$ to be the smallest difference between the signal $\mathbf{x}$ and the best model-based approximation $\hat{\mathbf{x}} \in \mathcal{M}_K$, i.e.

$$\sigma_{\mathcal{M}_K} \triangleq \inf_{\hat{\mathbf{x}} \in \mathcal{M}_K} \|\mathbf{x} - \hat{\mathbf{x}}\|_2 = \|\mathbf{x} - \mathbb{M}(\mathbf{x}, K)\|_2. \tag{5.105}$$

If this approximation error decays according to a power law (in the model size) we class this as a model compressible signal.

Definition 5.18. *The set of s-model compressible signals is defined to be*

$$\mathfrak{M}_s \triangleq \left\{ \mathbf{x} \in \mathbb{R}^n : \sigma_{\mathcal{M}_k}(\mathbf{x}) \leqslant SK^{-1/s}, \, 1 \leqslant K \leqslant n, S < \infty \right\}. \tag{5.106}$$

Any signal $\mathbf{x}$ in this set we say is s-model compressible under the signal model $\mathcal{M}_K$. Define $|\mathbf{x}|_{\mathfrak{M}_S}$ as the smallest value of S for which this condition holds for $\mathbf{x}$ and s.

The class of signals that are model compressible is much larger than the associated class of sparse signals. In conventional compressive sensing we have the same sufficient condition on the measurement matrix Φ for the stable recovery of both sparse and compressible signals. But because of the extra size of model compressible signals, we require further conditions to ensure the stable recovery of signals for model based compressive sensing. In particular we are interested in models that generate nested approximations, i.e. they obey the *nested approximation property*.

Definition 5.19. *A model $\mathcal{M} = \{\mathcal{M}_1, \mathcal{M}_2, \ldots\}$ obeys the* nested approximation property *or* NAP *if the approximations of different size it produces are nested, that is, if*

$$\text{supp}(\mathbb{M}(\mathbf{x}, K)) \subset \text{supp}(\mathbb{M}(\mathbf{x}, K')) \qquad \textit{for all } K < K', \mathbf{x} \in \mathbb{R}^n. \tag{5.107}$$

For models obeying the NAP we can define the residual in going from one model size to a larger model size. Let $\mathcal{R}_{j,K}(\mathcal{M})$ be the j-th set of residual subspaces given by

$$\mathcal{R}_{j,K}(\mathcal{M}) = \left\{ \mathbf{u} \in \mathbb{R}^n : \mathbf{u} = \mathbb{M}(\mathbf{x}, jK) - \mathbb{M}\left(\mathbf{x}, (j-1)K\right), \mathbf{x} \in \mathbb{R}^n \right\}. \tag{5.108}$$

Finally we can introduce the generalisation of the RIP, the *restricted amplification property*.

Definition 5.20. *A matrix* $\Phi \in \mathbb{R}^{m \times n}$ *has the* (δ_K, r)*-restricted amplification property or RAmP for the residual subspaces* $\mathcal{R}_{j,K}$ *of model* $\mathcal{M}$ *if*

$$\|\Phi \mathbf{u}\|_2^2 \leqslant (1 + \delta_K) j^{2r} \|\mathbf{u}\|_2^2, \tag{5.109}$$

for any $\mathbf{u} \in \mathcal{R}^{j,K}$ *for each* $1 \leqslant j \leqslant \lceil n/K \rceil$.

Similar to the RIP we can say how large we require m, the number of measurements to be, for a matrix $\Phi \in \mathbb{R}^{m \times n}$ to obey the RAmP.

Theorem 5.21 (Theorem 2 of (Baraniuk, Cevher, Duarte, and Hegde 2008)). *Let* $\Phi \in \mathbb{R}^{m \times n}$ *be generated from a subgaussian distribution and let the set of residual subspaces* $\mathcal{R}_{j,K}$ *of model* $\mathcal{M}$ *contain* R_j *subspaces of dimension K for each* $1 \leqslant j \leqslant \lceil n/K \rceil$. *Then, if*

$$m \geqslant \max_{1 \leqslant j \leqslant \lceil n/K \rceil} \frac{2K + 4 \log \frac{R_j n}{K} + 2t}{\left(j^r \sqrt{1 + \delta_K} - 1 \right)^2}, \tag{5.110}$$

the matrix Φ *has the* (δ_K, r)*-RAmP with probability at least* $1 - e^{-t}$.

Using the restricted amplification property we can then show how well model based CS allows us to reconstruct s-model compressible signals.

Theorem 5.22 (Theorem 3 of (Baraniuk, Cevher, Duarte, and Hegde 2008)). *Let* $\mathbf{x} \in \mathfrak{M}_s$ *be an s-model compressible signal under the model* $\mathcal{M}$ *that obeys the NAP. Furthermore let* $\Phi \in \mathbb{R}^{m \times n}$ *have the* (δ_K, r)*-restricted amplification property for* $r = s - 1$. *Then we have*

$$\|\Phi \mathbf{x} - \Phi \mathbb{M}(\mathbf{x}, K)\|_2 \leqslant K^{-s} \sqrt{1 + \delta_K} \log \left\lceil \frac{n}{K} \right\rceil |\mathbf{x}|_{\mathfrak{M}_s}. \tag{5.111}$$

Observe that Theorem 5.22 provides a bound on the difference in Φ-space of a compressible signal and its best model based approximation.

In sections 5.3.1 and 5.3.2 we will show how we can modify the existing CoSaMP and IHT algorithms to perform model based compressive sensing.

5.3.1   Model based CoSaMP

Modifying the CoSaMP algorithm to perform model based compressive sensing is relatively easy. Essentially it involves two modifications to Algorithm 5.4 in steps 7 and 10. Instead of taking the best s and $2s$ sparse approximations, we simply take the best model based approximations at these steps, which can be seen in Algorithm 5.8.

In Figure 5.1 we graphically illustrate the relative performance of model based and non-model based CoSaMP, taken from (Baraniuk, Cevher, Duarte, and Hegde 2008).

We can now illustrate the performance of the new model based CoSaMPalgorithm. Theorem 5.23 upper bounds the error when dealing with model sparse signals and Theorem 5.24 upper bounds the error for model compressible signals.

Theorem 5.23 (Theorem 4 of (Baraniuk, Cevher, Duarte, and Hegde 2008)). *Let* $\mathbf{x} \in \mathcal{M}_s$ *and let* $\mathbf{y} = \Phi \mathbf{x} + \mathbf{e}$ *where* $\mathbf{x}, \mathbf{e} \in \mathbb{R}^n$, $\mathbf{y} \in \mathbb{R}^m$ *and* $\Phi \in \mathbb{R}^{m \times n}$ *be a set of noisy compressive sensing*

Figure 5.1: Example performance of model based and standard CoSaMP recovery. From left to right, the original image with $n = 128 \times 128 = 16,384$ pixels and the images recovered using $m = 5,000$ measurements from the standard CoSaMP algorithm (middle) and the model based recovery (right). Taken from (Baraniuk, Cevher, Duarte, and Hegde 2008, Fig. 5).

measurements. Then if Φ has an $\mathcal{M}_s^4$-RIP constant of $\delta_{\mathcal{M}_s^4} \leqslant 0.1$, then the signal estimate $\mathbf{x}^{(k)}$ obtained at iteration k of the model based CoSaMP algorithm satisfies

$$\left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2 \leqslant 2^{-k} \left\| \mathbf{x} \right\|_2 + 15 \left\| \mathbf{e} \right\|_2 . \tag{5.112}$$

Theorem 5.24 (Theorem 5 of (Baraniuk, Cevher, Duarte, and Hegde 2008))**.** *Let $\mathbf{x} \in \mathfrak{M}_S$ be an S-model compressible signal from a model $\mathcal{M}$ that obeys the NAP and let $\mathbf{y} = \Phi\mathbf{x} + \mathbf{e}$ where $\mathbf{x}, \mathbf{e} \in \mathbb{R}^n$, $\mathbf{y} \in \mathbb{R}^m$ and $\Phi \in \mathbb{R}^{m \times n}$ be a set of noisy compressive sensing measurements. Then if Φ has an $\mathcal{M}_s^4$-RIP constant of $\delta_{\mathcal{M}_s^4} \leqslant 0.1$ and the (δ_s, r)-RAmP with $\delta_s \leqslant 0.1$ and $r = S - 1$, then the signal estimate $\mathbf{x}^{(k)}$ at the k-th iteration of the model based CoSaMP algorithm satisfies*

$$\left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2 \leqslant 2^{-k} \left\| \mathbf{x} \right\|_2 + 35 \left(\left\| \mathbf{e} \right\|_2 + |\mathbf{x}|_{\mathfrak{M}_S} s^{-S} \left(1 + \log \left\lceil \frac{n}{s} \right\rceil \right) \right) . \tag{5.113}$$

Another important question is what happens in the case of a model mismatch, that is, when the measurements obtained do not exactly conform to the model being used to reconstruct the data? As in the original paper, let us first consider the scenario where we are using a model that is "close" to the true model. Assume that out signal $\mathbf{x}$ is not s-model sparse but rather $s + \varsigma$-model sparse for some small integer ς. Then if the matrix $\Phi \in \mathbb{R}^{m \times n}$ has the (δ_s, r)-RAmP then it follows that

$$\left\| \Phi(\mathbf{x} - \mathbf{x}^s) \right\|_2 \leqslant 2^r \sqrt{1 + \delta_s} \left\| \mathbf{x} - \mathbf{x}^s \right\|_2 . \tag{5.114}$$

Using this and Theorem 5.23 we get the following bound

$$\left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2 \leqslant 2^{-k} \left\| \mathbf{x} \right\|_2 + 16 \cdot 2^r \sqrt{1 + \delta_s} \left\| \mathbf{x} - \mathbf{x}^s \right\|_2 + 15 \left\| \mathbf{e} \right\|_2 . \tag{5.115}$$

This is larger only by the amount $16 \cdot 2^r \sqrt{1 + \delta_s} \left\| \mathbf{x} - \mathbf{x}^s \right\|_2$ than in Theorem 5.23, and if the s-sparse approximation is good, this term will be small.

Secondly, consider the model compressible case. Let $\mathbf{x}$ be a signal that is not S-model compressible but rather $(S - \varepsilon)$-model compressible. Then in the k-th iteration we obtain the bound

$$\left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2 \leqslant 2^{-k} \left\| \mathbf{x} \right\|_2 + 35 \left(\left\| \mathbf{e} \right\|_2 + |\mathbf{x}|_{\mathfrak{M}_S} s^{-S} \left(1 + \frac{\left\lceil \frac{n}{k} \right\rceil^\varepsilon - 1}{\varepsilon} \right) \right) , \tag{5.116}$$

which is proved in Appendix II of (Baraniuk, Cevher, Duarte, and Hegde 2008). Observe that as $\varepsilon \to 0$, this upper bound on the error approaches that of Theorem 5.24.

Algorithm 5.8 Model based CoSaMP

Input:

- ▶ s the sparsity of $\mathbf{x}$
- ▶ $\mathbf{y} \in \mathbb{R}^m$ and the measurement matrix $\Phi \in \mathbb{R}^{m \times n}$
- ▶ A model $\mathcal{M}_s$

Output:

- ▶ $\hat{\mathbf{x}} \in \mathbb{R}^n$ such that $\hat{\mathbf{x}}$ is s-sparse, conforms to the model $\mathcal{M}_s$ and $\mathbf{y} = \Phi \hat{\mathbf{x}}$

1: $\mathbf{x}^{(0)} \leftarrow \mathbf{0}$
2: $\mathbf{v} \leftarrow \mathbf{y}$
3: $k \leftarrow 0$
4: **while** Halting condition false **do**
5: $\quad k \leftarrow k+1$
6: $\quad \mathbf{z} \leftarrow \Phi^\mathsf{T}\mathbf{v}$ ▷ Signal proxy
7: $\quad \Omega \leftarrow \text{supp}\left(\mathbb{M}_2(\mathbf{z},s)\right)$ ▷ Support of model based residual approximation
8: $\quad \Gamma \leftarrow \Omega \cup \text{supp}\left(\mathbf{x}^{(k-1)}\right)$ ▷ Merge supports
9: $\quad \bar{\mathbf{x}} \leftarrow \arg\max_{\tilde{\mathbf{x}}:\text{supp}(\tilde{\mathbf{x}})=\Gamma} \left\|\Phi\tilde{\mathbf{x}} - \mathbf{y}\right\|_2$ ▷ Solve Least Squares
10: $\quad \mathbf{x}^{(k)} \leftarrow \mathbb{M}(\bar{\mathbf{x}},s)$ ▷ Prune according to signal model
11: $\quad \mathbf{v} \leftarrow \mathbf{y} - \Phi\mathbf{x}^{(k)}$ ▷ Update current sample
12: **end while**
13: $\hat{\mathbf{x}} \leftarrow \mathbf{x}^{(k)}$
14: **return** $\hat{\mathbf{x}}$

5.3.2 Model based IHT

Similar to the CoSaMP algorithm, to incorporate model based CS into the IHT algorithm we only need to replace the pruning step with a model based pruning method. We present the model based version of Algorithm 5.5, the original iterative hard thresholding algorithm as Algorithm 5.9. We can apply Theorems 5.23 and 5.24 from before with only minor modifications:

- ▶ Φ must obey the $\mathcal{M}_K^3$-RIP (not $\mathcal{M}_K^4$-RIP) with constant $\delta_{\mathcal{M}_K^3} \leqslant 0.1$,

- ▶ the constant in front of the $\mathbf{e}$ term in Theorem 5.23 changes from 15 to 4, i.e. the estimate $\mathbf{x}^{(k)}$ at the k-th iteration obeys

$$\left\|\mathbf{x} - \mathbf{x}^{(k)}\right\|_2 \leqslant 2^{-k}\left\|\mathbf{x}\right\|_2 + 4\left\|\mathbf{e}\right\|_2. \tag{5.117}$$

- ▶ the constant factor in Theorem 5.24 changes from 35 to 10, i.e. the estimate $\mathbf{x}^{(k)}$ at the k-th iteration obeys

$$\left\|\mathbf{x} - \mathbf{x}^{(k)}\right\|_2 \leqslant 2^{-k}\left\|\mathbf{x}\right\|_2 + 10\left(\left\|\mathbf{e}\right\|_2 + |\mathbf{x}|_{\mathfrak{M}_S}\, s^{-S}\left(1 + \log\left\lceil\frac{n}{s}\right\rceil\right)\right). \tag{5.118}$$

In Figure 5.2 we give an illustration of how model based iterative hard thresholding can outperform iterative hard thresholding. This example was taken from (Baraniuk, Cevher, Duarte, and Hegde 2008, Fig. 7)

Algorithm 5.9 Model Based Iterative Hard Thresholding

Input:

- ▶ s the sparsity of $\mathbf{x}$
- ▶ $\mathbf{y} \in \mathbb{R}^m$ and the measurement matrix $\Phi \in \mathbb{R}^{m \times n}$
- ▶ model based pruning function $\mathbb{M}(\hat{\mathbf{x}}, s)$

Output:

- ▶ $\hat{\mathbf{x}} \in \mathbb{R}^n$ such that $\hat{\mathbf{x}}$ is s sparse and $\mathbf{y} = \Phi\hat{\mathbf{x}}$

1: $\mathbf{x}^{(0)} \leftarrow \mathbf{0}$
2: **for** $i = 1, \ldots$ **do**
3: $\quad \mathbf{x}^{(i)} \leftarrow \mathbb{M}\left(\mathbf{x}^{(i-1)} + \Phi^\mathsf{T}\left(\mathbf{y} - \Phi\mathbf{x}^{(i-1)}\right), s\right)$
4: **end for**
5: $\hat{\mathbf{x}} \leftarrow \mathbf{x}^{(i)}$
6: **return** $\hat{\mathbf{x}}$
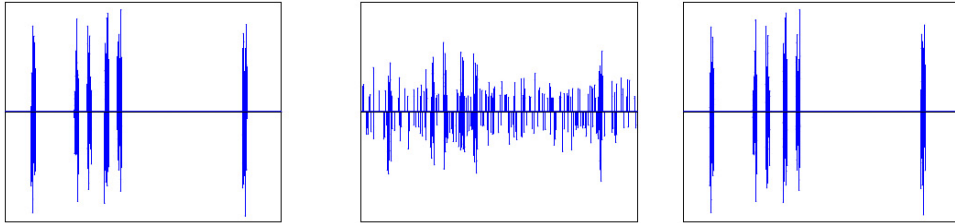
Figure 5.2: Example performance of model based and standard iterative hard thresholding recovery. From left to right, the original piecewise smooth `HeaviSine` test signal with $n = 1024$ and the signals recovered using $m = 80$ from the standard recovery algorithm (middle) and the model based recovery (right). Taken from (Baraniuk, Cevher, Duarte, and Hegde 2008, Fig. 7)

5.4 Iteratively Re-weighted Least Squares Minimisation

Iteratively Re-weighted Least Squares Minimisation or IRLS algorithms offer yet another alternative to directly solving the ℓ_1 minimisation problem. The basis for these algorithms stems from the following theorem.

Theorem 5.25 ((Daubechies, DeVore, Fornasier, and Gunturk 2008)). *Let $\hat{\mathbf{x}} \in \mathbb{R}^n$ be the solution to the ℓ_1 problem*

$$\hat{\mathbf{x}} = \arg\min \|\mathbf{x}\|_1 \qquad \textit{subject to} \qquad \mathbf{y} = \Phi\mathbf{x}, \tag{5.119}$$

where $\mathbf{y} \in \mathbb{R}^m$ and $\Phi \in \mathbb{R}^{m \times n}$. Then if Equation (5.119) has a solution $\hat{\mathbf{x}}$ with no coordinates equal to zero, then the unique solution $\mathbf{x}^{\mathbf{w}}$ of the weighted least squares problem

$$\mathbf{x}^{\mathbf{w}} = \arg\min \|\mathbf{x}\|_{\ell_2(\mathbf{w})} \quad \textit{subject to} \quad \mathbf{y} = \Phi\mathbf{x}, \tag{5.120}$$

where

$$\mathbf{w} \triangleq \left(\frac{1}{|\hat{x}_1|}, \ldots, \frac{1}{|\hat{x}_n|}\right) \in \mathbb{R}^n, \tag{5.121}$$

is identical to $\hat{\mathbf{x}}$.

Proof. We prove the theorem by contradiction. Assume that $\hat{\mathbf{x}}$, the solution to Equation (5.119) is not the solution to Equation (5.120), i.e. not the $\ell_2(\mathbf{w})$ minimiser. Then there exists $\mathbf{n} \in \ker(\Phi)$

such that

$$\|\hat{\mathbf{x}} + \mathbf{n}\|_{\ell^2(\mathbf{w})} < \|\hat{\mathbf{x}}\|_{\ell^2(\mathbf{w})} \tag{5.122a}$$

$$\Longleftrightarrow \quad \frac{1}{2}\|\mathbf{n}\|_{\ell^2(\mathbf{w})} < -\sum_{j=1}^{n} w_j n_j \hat{x}_j = \sum_{j=1}^{n} n_j \operatorname{sign}(\hat{x}_j). \tag{5.122b}$$

However $\hat{\mathbf{x}}$ is an ℓ_1 minimiser, so for $\mathbf{n} \in \ker(\Phi)$ we must have

$$\|\hat{\mathbf{x}}\|_1 \leqslant \|\hat{\mathbf{x}} + \varepsilon\mathbf{n}\|_1, \qquad \varepsilon \neq 0, \tag{5.123}$$

hence taking ε sufficiently small implies

$$\sum_{j=1}^{n} n_j \operatorname{sign}(\hat{x}_j) = 0, \tag{5.124}$$

which is a contradiction. $\qquad\square$

There are two issues with applying this iterative approach to finding the ℓ_1 minimiser directly. Firstly we do not know $\hat{\mathbf{x}}$ and secondly in the compressed sensing paradigm, we assume that the original signal is sparse, invalidating the original assumption that the components of $\mathbf{x}$ are non-zero.

The first issue was addressed in (Lawson 1961) by choosing starting weights $\mathbf{w}^{(0)}$ and solving Equation (5.120) for this weight. This solution is then used to create new weights $\mathbf{w}^{(1)}$ and Equation (5.120) is solved again but with the new weights, and so on. This algorithm is now known as *Lawson's algorithm* and can be extended to the solution of other ℓ_p minimisation problems. It is shown in (Osborne 1985) that for $1 < p < 3$ and the appropriate IRLS algorithm, the algorithm converges to the correct solution. For $p = 1$, using the weight update (at the i-th iteration)

$$w_j^{(i+1)} = \frac{1}{\left|\hat{x}_j^{(i)}\right|}, \qquad i = 1, \dots, n, \tag{5.125}$$

the algorithm will converge if Equation (5.119) has a unique solution (and all components are non-zero). Furthermore the rate of convergence is linear. Problems still occur if one of the components of $\hat{\mathbf{x}}$ vanish as the corresponding weight will go to infinity.

The paper (Daubechies, DeVore, Fornasier, and Gunturk 2008) addresses the issue of the weights going to zero when a component of $\hat{\mathbf{x}}$ is zero. Their algorithm can be seen in Algorithm 5.10. We first need to define the function $\mathcal{J}(\mathbf{z}, \mathbf{w}, \varepsilon)$

$$\mathcal{J}(\mathbf{z}, \mathbf{w}, \varepsilon) \triangleq \frac{1}{2}\sum_{j=1}^{n}\left(z_j^2 w_j + \varepsilon^2 w_j + \frac{1}{w_j}\right), \tag{5.126}$$

for $\mathbf{z}, \mathbf{w} \in \mathbb{R}^n$ and $\varepsilon > 0$. Note that this function is strictly convex. Let us also define the function $r : \mathbb{R}^n \to \mathbb{R}^n$ such that $r(\mathbf{z})$ returns the components of $\mathbf{z}$ sorted in non-increasing order by magnitude, hence $r(\mathbf{z})_j$ is the j-th largest component of $\mathbf{z}$ sorted by absolute value.

One can rewrite the solution and weight update of Algorithm 5.10 going to iteration $i + 1$ in

---

Algorithm 5.10 Iteratively Re-weighted Least Squares Minimisation

---

Input:
Output:
1: $\mathbf{w}^{(0)} \leftarrow \mathbf{1}$
2: $\varepsilon^{(0)} \leftarrow 1$
3: $i \leftarrow 0$
4: **while** $\varepsilon_i \neq 0$ **do**
5:    $\mathbf{x}^{(i+1)} \leftarrow \arg\min_{\mathbf{z}} \mathcal{J}\left(\mathbf{z}, \mathbf{w}^{(i)}, \varepsilon^{(i)}\right)$ subject to $\mathbf{y} = \Phi\mathbf{z}$
6:    $\varepsilon^{(i+1)} \leftarrow \min\left(\varepsilon^{(i)}, \frac{r(\mathbf{x}^{i+1})_{K+1}}{n}\right)$ ▷ for some K
7:    $\mathbf{w}^{(i+1)} \leftarrow \arg\min_{\mathbf{w}>\mathbf{0}} \mathcal{J}\left(\mathbf{x}^{(i+1)}, \mathbf{w}, \varepsilon^{(i+1)}\right)$
8:    $i \leftarrow i+1$
9: **end while**

---

closed form

$$\mathbf{x}^{(i+1)} \leftarrow D_i \Phi^{\mathsf{T}} \left(\Phi D_i \Phi^{\mathsf{T}}\right)^{-1} \mathbf{y}, \tag{5.127a}$$

$$w_j^{(i+1)} \leftarrow \frac{1}{\sqrt{\left(x_j^{(i+1)}\right)^2 + \left(\varepsilon^{(i+1)}\right)^2}}, \qquad \text{for } j = 1, \ldots, n, \tag{5.127b}$$

where D_n is the $n \times n$ diagonal matrix with the weights $\mathbf{w}^{(i)}$ along the diagonal.

The main Theorem of (Daubechies, DeVore, Fornasier, and Gunturk 2008), Theorem 5.26 regards the convergence of Algorithm 5.10. First we make the following definitions. Let Σ_k be the set of all vectors in $\mathbb{R}^n$ that have at most k non-zero entries. Then for any $\mathbf{z} \in \mathbb{R}^n$ define

$$\sigma_j(\mathbf{z})_{\ell_1} \triangleq \inf_{\mathbf{x} \in \Sigma_k} \|\mathbf{z} - \mathbf{x}\|_1, \tag{5.128}$$

which is the smallest approximation error in approximating the vector $\mathbf{z}$ by a k-sparse vector.

Theorem 5.26 (Theorem 5.3 of (Daubechies, DeVore, Fornasier, and Gunturk 2008))**.** *Let K (of the update rule for ε_i in Algorithm 5.10) be chosen so that Φ satisfies the Null Space Property of order K for some $\gamma < 1$. Then, for each $\mathbf{y} \in \mathbb{R}^m$, the output of Algorithm 5.10 converges to a vector $\hat{\mathbf{x}}$, with $r(\hat{\mathbf{x}})_{K+1} = n \lim_{i \to \infty} \varepsilon_i$ and the following hold:*

(a) *If $\varepsilon = \lim_{i \to \infty} \varepsilon_i$, the $\hat{\mathbf{x}}$ is K-sparse; in this case there is a unique ℓ_1 minimiser $\mathbf{x}^*$ and $\hat{\mathbf{x}} = \hat{\mathbf{x}}^*$. Moreover we have, for $k \leqslant K$ and any $\mathbf{z}$ such that $\mathbf{y} = \Phi\mathbf{z}$*

$$\|\mathbf{z} - \hat{\mathbf{x}}\|_1 \leqslant c\sigma_k(\mathbf{z})_{\ell_1}, \qquad \text{with } c \triangleq \frac{2(1+\gamma)}{1-\gamma}. \tag{5.129}$$

(b) *If $\varepsilon = \lim_{i \to \infty} \varepsilon_i > 0$ then $\hat{\mathbf{x}} = \mathbf{x}^\varepsilon$.*

(c) *If γ satisfies the tighter bound $\gamma < 1 - \frac{2}{K+2}$ then for all $\mathbf{z}$ such that $\mathbf{y} = \Phi\mathbf{z}$ and any $k < K - \frac{2\gamma}{1-\gamma}$, that*

$$\|\mathbf{z} - \hat{\mathbf{x}}\|_1 \leqslant \tilde{c}\sigma_k(\mathbf{z})_{\ell_1}, \qquad \text{with } \tilde{c} \triangleq \frac{2(1+\gamma)}{1-\gamma} \cdot \frac{K - k + \frac{3}{2}}{K - k - \frac{2\gamma}{1-\gamma}}. \tag{5.130}$$

This case does not occur if there exists a vector $\mathbf{z}$ where $\mathbf{y} = \Phi\mathbf{z}$ that has sparsity $k < K - \frac{2\gamma}{1-\gamma}$.

5.5   Belief Propagation Decoding

A rather different algorithm to the previously mentioned ones, *Belief Propagation* or *BP* is a message passing algorithm. This type of algorithm has commonly been used for error correction decoding when it was demonstrated in (Gallager 1963) that they could be used for fast decoding of *low-density parity check* (LDPC) codes. The difficult step in decoding LDPC codes is the reconstruction of the codeword given the received message, i.e. finding the "best" $\mathbf{x}$ given a $\mathbf{y}$ where $\mathbf{y} = G\mathbf{x} + \mathbf{e}$ so that the error $\mathbf{e}$ is minimal in some sense, $G \in \mathbb{R}^{n \times k}$ is the *generating matrix* for $n \geqslant k$. In some ways this is the reverse of compressed sensing as we are trying to solve an over-constrained problem, but nonetheless recent work in (Sarvotham, Baron, and Baraniuk 2006) shows that this technique can be used for compressive sensing reconstruction. The algorithm presented in this paper is not a generic algorithm, it requires a specific form for the measurement matrix Φ, namely that Φ has entries restricted to $\{-1, 0, 1\}$ and is sparse. It is also highly desirable that the matrix Φ is cycle-free or at least contains no cycles of small degree.

Because of the restrictions of this decoding algorithm, and that there are, to the best of my knowledge, no theoretical results regarding its performance, we will not present the algorithm, but we mention it for completeness.

5.6   Comparison of Algorithms

We will give a discussion and comparison of these reconstruction algorithms in Chapter 7, including a simulation of some of the discussed methods.

6 Modified Frame Reconstruction Algorithm

In this chapter we will present a new reconstruction algorithm for compressive sensing. The "Modified Frame Reconstruction algorithm" or MFR is motivated by the frame reconstruction algorithm of Chapter 4. This work generalises previously known results, such as the iterative hard thresholding (IHT) algorithm in (Blumensath and Davies 2008), although our work was developed independently and without knowledge of this paper.

We give the plain version of the MFR algorithm and then show two modifications both of which increase the rate of convergence and the success rate in finding the original sparse vector. We will also show theoretical bounds for the convergence and give sufficient conditions for convergence to occur.

6.1 Background

Let $\{\psi_i\}$, $i = 1, \ldots, t$ for some $t \geqslant n$ form a frame for the n-dimensional Hilbert space $\mathscr{H}$ and let the frame have frame bounds $0 < A \leqslant B < \infty$, that is

$$A \left\| \mathbf{x} \right\|^2 \leqslant \sum_{i=1}^{t} \left| \langle \mathbf{x}, \psi_i \rangle \right|^2 \leqslant B \left\| \mathbf{x} \right\|^2, \tag{6.1}$$

for all $\mathbf{x} \in \mathscr{H}$. Then given a sequence of inner-products $\mathbf{y} \in \mathbb{R}^t$ where $y_i = \langle \mathbf{x}, \psi_i \rangle$, the fundamental part of the iterative reconstruction algorithm is the update

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \frac{2}{A + B} \sum_{j=1}^{n} \psi_j \left(\mathbf{y}_j - \left\langle \psi_j, \mathbf{x}^{(k)} \right\rangle \right). \tag{6.2}$$

In particular if $\mathscr{H}$ is the Hilbert space $\mathbb{R}^n$ then we can rewrite this in matrix form, where Ψ is the matrix that has the frame elements as rows, to get the update

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \frac{2}{A+B}\Psi^{\mathsf{T}}\left(\mathbf{y} - \Psi\mathbf{x}^{(k)}\right). \tag{6.3}$$

Observe that Equation (6.3) converges to the solution $\hat{\mathbf{x}}$ of the matrix equation $\mathbf{y} = \Psi\hat{\mathbf{x}}$, given the inner-products $\mathbf{y}$ and the matrix Ψ. Furthermore the frame bounds are given by the largest and smallest eigenvalues of $\Psi^{\mathsf{T}}\Psi$.

The key observation for MFR is the fact that we can still perform this iterative step even if $t < n$, that is, the matrix Ψ no longer forms a frame for the space $\mathbb{R}^n$. And that if the algorithm converges, it can still converge to a solution $\hat{\mathbf{x}}$ of the now under-determined matrix equation $\mathbf{y} = \Psi\hat{\mathbf{x}}$, which is exactly the problem of compressive sensing.

6.2 Algorithm

We present here the basic version of the MFR algorithm. The algorithm consists of two parts, an update and a thresholding step.

(a) **Update:** Similar to the frame algorithm we perform an update

$$\mathbf{a}^{(k+1)} = \mathbf{x}^{(k)} + \gamma\Phi^{\mathsf{T}}\left(\mathbf{y} - \Phi\mathbf{x}^{(k)}\right), \tag{6.4}$$

where $\mathbf{y}$ is the vector of measurements, Φ is the measurement matrix and γ is a control parameter which we refer to as the *step-length*.

(b) **Thresholding:** The second part of the algorithm is the thresholding procedure where we generate the next "solution"

$$\mathbf{x}^{(k+1)} = \mathbb{H}_{\hat{s}}(\mathbf{a}^{(k+1)}). \tag{6.5}$$

Here we simply threshold the output of the frame step producing an $\hat{s}$-sparse approximation.

Recall that $\mathbb{H}_{\hat{s}}(\mathbf{z})$ produces the best $\hat{s}$-sparse approximation to the input $\mathbf{z}$ under any ℓ_p norm for $1 \leqslant p < \infty$, shown in Lemma 2.14. These two steps are repeated until some criteria is met for termination, typically one would stop the algorithm once "convergence" is achieved, namely that the change from one iteration to the next is small, i.e. $\left\|\mathbf{x}^{(k)} - \mathbf{x}^{(k+1)}\right\|_2 \leqslant \varepsilon$ for some tolerance level ε.

Formally we present the algorithm as Algorithm 6.1.

One important question is what value of γ should we use. If we take inspiration from the frame reconstruction algorithm we should set $\gamma = 2/(A+B)$ where A and B are the frame bounds. Since we are dealing only with sparse vectors we can use tighter frame bounds, where A and B are the frame bounds for $(s+\hat{s})$-sparse vectors, namely the largest value A and the minimal B such that

$$A\|\mathbf{x}\|^2 \leqslant \sum_n |\langle\mathbf{x}, \psi_n\rangle|^2 \leqslant B\|\mathbf{x}\|^2, \tag{6.6}$$

Algorithm 6.1 Modified Frame Reconstruction Algorithm

Input:

- ▶ The measurement matrix Φ.
- ▶ Observation vector $\mathbf{y}$.
- ▶ Estimate of sparsity $\hat{s}$ of the vector $\mathbf{x}$.
- ▶ Step-length γ.

Output:

- ▶ A solution $\hat{\mathbf{x}}$ to $\mathbf{y} = \Phi\mathbf{x}$.

1: $\mathbf{x}^{(0)} \leftarrow 0$
2: **for** $k = 0, \ldots$ **do**
3:     $\mathbf{x}^{(k+1)} \leftarrow \mathbb{H}_{\hat{s}}\left(\mathbf{x}^{(k)} + \gamma\Phi^\mathsf{T}\left(\mathbf{y} - \Phi\mathbf{x}^{(k)}\right)\right)$
4: **end for**
5: **return** $\mathbf{x}^{(k+1)}$

---

for all $(s + \hat{s})$-sparse vectors $\mathbf{x}$. In particular, from the RIP we have

$$1 - \delta_{s+\hat{s}} \leqslant A \leqslant B \leqslant 1 + \delta_{s+\hat{s}}, \tag{6.7}$$

with equality on at least one of the sides, i.e. $A = 1 - \delta_{s+\hat{s}}$ or $B = 1 + \delta_{s+\hat{s}}$, but not necessarily both. This implies that we could take

$$\gamma = \frac{2}{1 - \delta_{s+\hat{s}} + 1 + \delta_{s+\hat{s}}} = 1 \approx \frac{2}{A + B}. \tag{6.8}$$

If we borrow terminology from the world of differential equations and numerical solutions, γ is the step-length and controls the rate of convergence. Typically increasing γ increases the rate of convergence at the cost of possibly stopping convergence altogether. On the other hand, by decreasing γ the algorithm is more likely to converge, but will converge more slowly. This can be seen in Figure 6.8.

Later on in this chapter we will discuss ways to select γ, including adaptive algorithms that produce a new "best" γ to use in every iteration. We will also discuss in Chapter 8 what happens if the RIP condition is not met, then the sparse singular values (RIP constants) are considerably more than 1, so using the idea from the frame algorithm we would have

$$\gamma = \frac{1}{\sigma_{min}^{\hat{s}} + \sigma_{max}^{\hat{s}}} \ll 1, \tag{6.9}$$

as $\sigma_{max}^{\hat{s}} > 1$.

6.3   Analysis of Performance

We now state the properties of the MFR algorithm in Theorem 6.1 and Proposition 6.2. Proposition 6.2 gives sufficient convergence conditions in the scenario where we measure a sparse signal with noise, i.e. in the model $\mathbf{y} = \Phi\mathbf{x} + \mathbf{e}$ where $\Phi \in \mathbb{R}^{m \times n}$ is the measurement matrix, $\mathbf{x} \in \mathbb{R}^n$ is the s-sparse data, $\mathbf{e} \in \mathbb{R}^m$ is noise and $\mathbf{y} \in \mathbb{R}^m$ is the observed signal. We will see in

Theorem 6.1 that the MFR algorithm converges even when the signal is not sparse.

The proof of these results is similar to the one in (Blumensath and Davies 2008).

Theorem 6.1. *Fix s the sparsity of the desired solution. For measurements* $\mathbf{y} = \Phi\mathbf{x} + \mathbf{e}$ *where* $\Phi \in \mathbb{R}^{m \times n}$ *has the RIP such that either condition (a), (b) or (c) is satisfied*

$$(a) \qquad\qquad \gamma \geqslant \frac{1}{\delta_{3s} - \delta_{2s} + 1}, \quad and \quad \gamma\delta_{3s} \leqslant \frac{1}{\sqrt{32}}, \qquad or \qquad\qquad (6.10a)$$

$$(b) \qquad\qquad \gamma < \frac{1}{\delta_{3s} - \delta_{2s} + 1}, \quad and \quad \gamma(1 - \delta_{2s}) \geqslant 1 - \frac{1}{\sqrt{32}}, \qquad or \qquad (6.10b)$$

$$(c) \qquad \frac{3}{4(1 - \delta_{2s})} < \gamma < \frac{1}{1 - \delta_{2s}} \quad and \quad \delta_{2s} < 1, \qquad\qquad\qquad (6.10c)$$

Algorithm 6.1 recovers an approximation $\mathbf{x}^{(k)}$ *satisfying*

$$\left\|\mathbf{x}^{(k)} - \mathbf{x}^s\right\|_2 \leqslant 2^{-k}\left\|\mathbf{x}^s\right\|_2 + 4\gamma\sqrt{1 + \delta_{2s}}\left\|\mathbf{e}\right\|_2 + 4\gamma(1 + \delta_{2s})\left(\left\|\mathbf{x} - \mathbf{x}^{2s}\right\|_2 + \frac{1}{\sqrt{2s}}\left\|\mathbf{x} - \mathbf{x}^{2s}\right\|_1\right),$$
$$(6.11)$$

where $\mathbf{x}^s$ *is the best s-sparse approximation to* $\mathbf{x}$*.*

Observe that the term $\left\|\mathbf{x} - \mathbf{x}^{2s}\right\|_2 + \frac{1}{\sqrt{2s}}\left\|\mathbf{x} - \mathbf{x}^{2s}\right\|_1$ is the unrecoverable energy used in (Needell and Tropp 2008; Blumensath and Davies 2008) which we made use of earlier.

Proposition 6.2. *Let* $\mathbf{x} \in \mathbb{R}^n$ *be s-sparse and fix the sparsity estimate* $\hat{s} \geqslant s$*. For measurements* $\mathbf{y} = \Phi\mathbf{x} + \mathbf{e}$ *where* $\Phi \in \mathbb{R}^{m \times n}$ *has the RIP such that either condition (a), (b) or (c) is satisfied*

$$(a) \qquad\qquad \gamma \geqslant \frac{1}{\delta_{s+2\hat{s}} - \delta_{s+\hat{s}} + 1}, \quad and \quad \gamma\delta_{s+2\hat{s}} \leqslant \frac{1}{\sqrt{32}}, \qquad or \qquad (6.12a)$$

$$(b) \qquad\qquad \gamma < \frac{1}{\delta_{s+2\hat{s}} - \delta_{s+\hat{s}} + 1}, \quad and \quad \gamma(1 - \delta_{s+\hat{s}}) \geqslant 1 - \frac{1}{\sqrt{32}}, \qquad or \qquad (6.12b)$$

$$(c) \qquad \frac{3}{4(1 - \delta_{s+\hat{s}})} < \gamma < \frac{1}{1 - \delta_{s+\hat{s}}} \quad and \quad \delta_{s+\hat{s}} < 1, \qquad\qquad (6.12c)$$

Algorithm 6.1 recovers an approximation $\mathbf{x}^{(k)}$ *satisfying*

$$\left\|\mathbf{x}^{(k)} - \mathbf{x}^s\right\|_2 \leqslant 2^{-k}\left\|\mathbf{x}^s\right\|_2 + 4\gamma\sqrt{1 + \delta_{s+\hat{s}}}\left\|\mathbf{e}\right\|_2. \qquad\qquad (6.13)$$

Proof. Conditions (a) and (b) come from Lemma 6.4 and condition (c) follows by setting $\alpha = \frac{1}{2}$ in Lemma 6.5. $\qquad\qquad\square$

We will first prove Lemmas 6.4 and 6.5, proving Proposition 6.2 and then use this to prove the theorem. To do this we require the following little lemma regarding recursive sequences.

Lemma 6.3. *Let* $v^{(k)} \leqslant \alpha v^{(k-1)} + \beta u$ *for* $u, v^{(k)} \geqslant 0$ *be a recursive sequence that holds for all positive integers k and for* α *such that* $|\alpha| < 1$*. Then*

$$v^{(k)} \leqslant \alpha^k v^{(0)} + \frac{\beta u}{1 - \alpha}. \qquad\qquad (6.14)$$

Proof. Observe that

$$v^{(k)} \leqslant \alpha v^{(k-1)} + \beta u \tag{6.15a}$$

$$\leqslant \alpha \left(\alpha v^{(k-2)} + \beta u \right) + \beta u \tag{6.15b}$$

$$\vdots$$

$$\leqslant \alpha^k v^{(0)} + \beta u \sum_{i=0}^{k} \alpha^i \tag{6.15c}$$

$$< \alpha^k v^{(0)} + \beta u \frac{1}{1-\alpha}, \tag{6.15d}$$

provided $|\alpha| < 1$. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \square$

Lemma 6.4. *Under the conditions of Proposition 6.2 and for measurements* $\mathbf{y} = \Phi\mathbf{x} + \mathbf{e}$ *where* $\Phi \in \mathbb{R}^{m \times n}$ *has the RIP such that only condition (a) or condition (b) is satisfied*

$$(a) \qquad \gamma \geqslant \frac{1}{\delta_{s+2\hat{s}} - \delta_{s+\hat{s}} + 1}, \quad and \quad \gamma \delta_{s+2\hat{s}} \leqslant \frac{1}{\sqrt{32}}, \qquad or \tag{6.16a}$$

$$(b) \qquad \gamma < \frac{1}{\delta_{s+2\hat{s}} - \delta_{s+\hat{s}} + 1}, \quad and \quad \gamma(1 - \delta_{s+\hat{s}}) \leqslant 1 - \frac{1}{\sqrt{32}}, \tag{6.16b}$$

Algorithm 6.1 recovers an approximation $\mathbf{x}^{(k)}$ *satisfying*

$$\left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2 \leqslant 2^{-k} \|\mathbf{x}\|_2 + 4\gamma\sqrt{1 + \delta_{s+\hat{s}}} \|\mathbf{e}\|_2 . \tag{6.17}$$

Proof of Lemma 6.4. First put

$$\mathbf{r}^{(i)} \triangleq \mathbf{x} - \mathbf{x}^{(i)}, \tag{6.18a}$$

$$\mathbf{a}^{(i)} \triangleq \mathbf{x}^{(i-1)} + \gamma\Phi^{\mathsf{T}}(\mathbf{y} - \Phi\mathbf{x}^{(i-1)}), \tag{6.18b}$$

$$\mathbf{x}^{(i)} \triangleq \mathbb{H}_{\hat{s}}(\mathbf{a}^{(i)}), \tag{6.18c}$$

$$\Gamma^{\star} \triangleq \mathrm{supp}(\mathbf{x}), \tag{6.18d}$$

$$\Gamma^{(i)} \triangleq \mathrm{supp}(\mathbf{x}^{(i)}), \tag{6.18e}$$

$$B^{(i)} \triangleq \Gamma^{\star} \cup \Gamma^{(i)}. \tag{6.18f}$$

As a consequence we have $|\Gamma^{\star}| \leqslant s$ and $\left|\Gamma^{(i)}\right| \leqslant \hat{s}$.

Consider the error $\left\| \mathbf{x}^s - \mathbf{x}^{(i+1)} \right\|_2$. Now we have $\mathbf{x} = \mathbf{x}^s = \mathbf{x}_{\Gamma^{\star}} = \mathbf{x}_{B^{(i)}}$ and $\mathbf{x}^{(i)} = \mathbf{x}^{(i)}_{\Gamma^{(i)}} = \mathbf{x}^{(i)}_{B^{(i)}}$. Although $\mathbf{x}$ is s-sparse, we will write $\mathbf{x}^s$ commonly to make the adaptation of this proof to the general case more obvious. Applying the triangle inequality we get

$$\left\| \mathbf{x}^s - \mathbf{x}^{(i+1)} \right\|_2 \leqslant \left\| \mathbf{x}^s_{B^{(i+1)}} - \mathbf{a}^{(i+1)}_{B^{(i+1)}} \right\|_2 + \left\| \mathbf{x}^{(i+1)}_{B^{(i+1)}} - \mathbf{a}^{(i+1)}_{B^{(i+1)}} \right\|_2 . \tag{6.19}$$

As $\mathbf{x}^{(i+1)}$ is the thresholded version of $\mathbf{a}^{(i+1)}$ it is the best s-term approximation to $\mathbf{a}^{(i+1)}$, in particular it is better than $\mathbf{x}^s$. Hence

$$\left\| \mathbf{x}^{(i+1)} - \mathbf{a}^{(i+1)}_{B^{(i+1)}} \right\|_2 \leqslant \left\| \mathbf{x}^s - \mathbf{a}^{(i+1)}_{B^{(i+1)}} \right\|_2 , \tag{6.20}$$

and thus Equation (6.19) becomes

$$\left\| \mathbf{x}^s - \mathbf{x}^{(i+1)} \right\|_2 \leqslant 2 \left\| \mathbf{x}^s_{B^{(i+1)}} - \mathbf{a}^{(i+1)}_{B^{(i+1)}} \right\|_2 . \tag{6.21}$$

Using the fact that $\mathbf{y} = \Phi \mathbf{x}^s + \mathbf{e}$ and $\mathbf{r}^{(i)} = \mathbf{x}^s - \mathbf{x}^{(i)}$ we get

$$\mathbf{a}^{(i+1)}_{B^{(i+1)}} = \mathbf{x}^{(i)}_{B^{(i+1)}} + \gamma \Phi^\mathsf{T}_{B^{(i+1)}} \left(\mathbf{y} - \Phi \mathbf{x}^{(i)}_{B^{(i+1)}} \right) \tag{6.22a}$$

$$= \mathbf{x}^{(i)}_{B^{(i+1)}} + \gamma \Phi^\mathsf{T}_{B^{(i+1)}} \Phi \mathbf{r}^{(i)} + \gamma \Phi^\mathsf{T}_{B^{(i+1)}} \mathbf{e}, \tag{6.22b}$$

hence

$$\left\| \mathbf{x}^s - \mathbf{x}^{(i+1)} \right\|_2 \leqslant 2 \left\| \mathbf{x}^s_{B^{(i+1)}} - \mathbf{x}^{(i)}_{B^{(i+1)}} - \gamma \Phi^\mathsf{T}_{B^{(i+1)}} \Phi \mathbf{r}^{(i)} - \gamma \Phi^\mathsf{T}_{B^{(i+1)}} \mathbf{e} \right\|_2 \tag{6.23a}$$

$$\leqslant 2 \left\| \mathbf{r}^{(i)}_{B^{(i+1)}} - \gamma \Phi^\mathsf{T}_{B^{(i+1)}} \Phi \mathbf{r}^{(i)} \right\|_2 + 2\gamma \left\| \Phi^\mathsf{T}_{B^{(i+1)}} \mathbf{e} \right\|_2 \tag{6.23b}$$

$$= 2 \left\| \left(\mathbf{I} - \gamma \Phi^\mathsf{T}_{B^{(i+1)}} \Phi_{B^{(i+1)}} \right) \mathbf{r}^{(i)}_{B^{(i+1)}} - \gamma \Phi^\mathsf{T}_{B^{(i+1)}} \Phi_{B^{(i)} \setminus B^{(i+1)}} \mathbf{r}^{(i)}_{B^{(i)} \setminus B^{(i+1)}} \right\|_2 + \cdots$$
$$+ 2\gamma \left\| \Phi^\mathsf{T}_{B^{(i+1)}} \mathbf{e} \right\|_2 \tag{6.23c}$$

$$\leqslant 2 \left\| \left(\mathbf{I} - \gamma \Phi^\mathsf{T}_{B^{(i+1)}} \Phi_{B^{(i+1)}} \right) \mathbf{r}^{(i)}_{B^{(i+1)}} \right\|_2 + 2\gamma \left\| \Phi^\mathsf{T}_{B^{(i+1)}} \Phi_{B^{(i)} \setminus B^{(i+1)}} \mathbf{r}^{(i)}_{B^{(i)} \setminus B^{(i+1)}} \right\|_2 + \cdots$$
$$+ 2\gamma \left\| \Phi^\mathsf{T}_{B^{(i+1)}} \mathbf{e} \right\|_2 , \tag{6.23d}$$

by repeated application of the triangle inequality and by splitting the residual into two parts, $\mathbf{r}^{(i+1)} = \mathbf{r}^{(i+1)}_{B^{(i+1)}} + \mathbf{r}^{(i+1)}_{B^{(i+1)} \setminus B^{(i)}}$. Then

$$\left| B^{(i)} \cup B^{(i+1)} \right| = \left| \Gamma^\star \cup \Gamma^{(i)} \cup \Gamma^{(i+1)} \right| \leqslant s + 2\hat{s}$$

as each set $\Gamma^{(k)}$ has at most $\hat{s}$ entries and $|\Gamma^\star| \leqslant s$. Recall from the RIP and Lemmas 3.10 and 3.11 that

$$\left\| \Phi^\mathsf{T}_\Lambda \mathbf{x} \right\|_2 \leqslant \sqrt{1 + \delta_s} \, \|\mathbf{x}\|_2 , \tag{6.24a}$$

$$\left\| \left(\mathbf{I} - \gamma \Phi^\mathsf{T}_\Lambda \Phi_\Lambda \right) \mathbf{x}_\Lambda \right\|_2 \leqslant (1 - \gamma(1 - \delta_s)) \, \|\mathbf{x}_\Lambda\|_2 , \tag{6.24b}$$

$$\left\| \Phi^\mathsf{T}_\Omega \Phi_{\Omega'} \mathbf{x}_{\Omega'} \right\|_2 \leqslant \delta_s \, \|\mathbf{x}_\Omega\|_2 , \tag{6.24c}$$

for all matrices Φ which obey the RIP and sets Λ, Ω, Ω', where Ω, Ω' are disjoint, $|\Lambda| = s$ and $|\Omega \cup \Omega'| = s$. We also have $\delta_s \leqslant \delta_{s'}$ for all positive integers $s \leqslant s'$. Applying Equation (6.24b) to the first term in Equation (6.23d), and applying Equations (6.24c) and (6.24a) to the second and third terms respectively, we get

$$\left\| \mathbf{r}^{(i+1)} \right\|_2 \leqslant 2 \left(1 - \gamma \left(1 - \delta_{s+\hat{s}} \right) \right) \left\| \mathbf{r}^{(i)}_{B^{(i+1)}} \right\|_2 + 2\gamma \delta_{s+2\hat{s}} \left\| \mathbf{r}^{(i)}_{B^{(i)} \setminus B^{(i+1)}} \right\|_2 + 2\gamma \sqrt{1 + \delta_{s+\hat{s}}} \, \|\mathbf{e}\|_2 . \tag{6.25}$$

The vectors $\mathbf{r}^{(i)}_{B^{(i+1)}}$ and $\mathbf{r}^{(i)}_{B^{(i)} \setminus B^{(i+1)}}$ are orthogonal as they have disjoint supports. Now let $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$ be two orthogonal vectors, then

$$\|\mathbf{u}\|_2 + \|\mathbf{v}\|_2 \leqslant \sqrt{2} \, \|\mathbf{u} + \mathbf{v}\|_2 . \tag{6.26}$$

We use this to bound the sum of the two terms $\left\| \mathbf{r}^{(i)}_{B^{(i+1)}} \right\|_2$ and $\left\| \mathbf{r}^{(i)}_{B^{(i)} \setminus B^{(i+1)}} \right\|_2$.

We first ask how do the terms $(1 - \gamma (1 - \delta_{s+\hat{s}}))$ and $\gamma \delta_{s+2\hat{s}}$ compare, given that $\delta_{s+\hat{s}} \leqslant \delta_{s+2\hat{s}}$. We then have either

$$1 - \gamma + \gamma \delta_{s+\hat{s}} \leqslant \gamma \delta_{s+2\hat{s}} \iff \gamma \geqslant \frac{1}{\delta_{s+2\hat{s}} - \delta_{s+\hat{s}} + 1}, \qquad \text{or} \qquad \text{(6.27a)}$$

$$1 - \gamma + \gamma \delta_{s+\hat{s}} > \gamma \delta_{s+2\hat{s}} \iff \gamma < \frac{1}{\delta_{s+2\hat{s}} - \delta_{s+\hat{s}} + 1}. \qquad \text{(6.27b)}$$

▶ **Case 1 – Equation** (6.27a): Equation (6.25) then becomes

$$\left\| \mathbf{r}^{(i+1)} \right\|_2 \leqslant 2\sqrt{2}\gamma \delta_{s+2\hat{s}} \left\| \mathbf{r}^{(i)} \right\|_2 + 2\gamma \sqrt{1 + \delta_{s+\hat{s}}} \left\| \mathbf{e} \right\|_2. \qquad \text{(6.28)}$$

Then if

$$2\sqrt{2}\gamma \delta_{s+2\hat{s}} \leqslant \frac{1}{2} \iff \gamma \delta_{s+2\hat{s}} \leqslant \frac{1}{\sqrt{32}}, \qquad \text{(6.29)}$$

we have

$$\left\| \mathbf{r}^{(i+1)} \right\|_2 \leqslant \frac{1}{2} \left\| \mathbf{r}^{(i)} \right\|_2 + 2\gamma \sqrt{1 + \delta_{s+\hat{s}}} \left\| \mathbf{e} \right\|_2. \qquad \text{(6.30)}$$

Applying the recursion using Lemma 6.3 we get

$$\left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2 \leqslant \frac{1}{2} \left\| \mathbf{r}^{(k-1)} \right\|_2 + 2\gamma \sqrt{1 + \delta_{s+\hat{s}}} \left\| \mathbf{e} \right\|_2 \qquad \text{(6.31a)}$$

$$\leqslant \frac{1}{2} \left(\frac{1}{2} \left\| \mathbf{r}^{(k-2)} \right\|_2 + 2\gamma \sqrt{1 + \delta_{s+\hat{s}}} \right) + 2\gamma \sqrt{1 + \delta_{s+\hat{s}}} \left\| \mathbf{e} \right\|_2 \qquad \text{(6.31b)}$$

$$\leqslant 2^{-k} \left\| \mathbf{x} - \mathbf{x}^{(0)} \right\|_2 + 2\gamma \sqrt{1 + \delta_{s+\hat{s}}} \left(\sum_{i=0}^{k} \frac{1}{2^i} \right) \left\| \mathbf{e} \right\|_2 \qquad \text{(6.31c)}$$

$$\leqslant 2^{-k} \left\| \mathbf{x} \right\|_2 + 4\gamma \sqrt{1 + \delta_{s+\hat{s}}} \left\| \mathbf{e} \right\|_2, \qquad \text{(6.31d)}$$

for all $k \geqslant 0$, since $\mathbf{x}^{(0)} = \mathbf{0}$ and provided that

$$\gamma \geqslant \frac{1}{\delta_{s+2\hat{s}} - \delta_{s+\hat{s}} + 1}, \qquad \text{and} \qquad \text{(6.32a)}$$

$$\gamma \delta_{s+2\hat{s}} \leqslant \frac{1}{\sqrt{32}}. \qquad \text{(6.32b)}$$

▶ **Case 2 – Equation** (6.27b): Equation (6.25) then becomes

$$\left\| \mathbf{r}^{(i+1)} \right\|_2 \leqslant 2\sqrt{2}(1 - \gamma(1 - \beta_{s+\hat{s}})) \left\| \mathbf{r}^{(i)} \right\|_2 + 2\gamma \sqrt{1 + \delta_{s+\hat{s}}} \left\| \mathbf{e} \right\|_2. \qquad \text{(6.33)}$$

If

$$2\sqrt{2}(1 - \gamma(1 - \delta_{s+\hat{s}})) \leqslant \frac{1}{2} \iff \gamma(\delta_{s+\hat{s}} - 1) \leqslant 1 - \frac{1}{\sqrt{32}} \qquad \text{(6.34a)}$$

$$\iff \delta_{s+\hat{s}} \leqslant 1 - \frac{1}{\gamma} + \frac{1}{\gamma\sqrt{32}} = \frac{8\gamma - 8 + \sqrt{2}}{8\gamma}, \qquad \text{(6.34b)}$$

we again have

$$\left\| \mathbf{r}^{(i+1)} \right\|_2 \leqslant \frac{1}{2} \left\| \mathbf{r}^{(i)} \right\|_2 + 2\gamma \sqrt{1 + \delta_{s+\hat{s}}} \left\| \mathbf{e} \right\|_2. \qquad \text{(6.35)}$$

Applying the recursion using Lemma 6.3 we get

$$\left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2 \leqslant 2^{-k} \left\| \mathbf{x} \right\|_2 + 4\gamma \sqrt{1 + \delta_{s+\hat{s}}} \left\| \mathbf{e} \right\|_2, \qquad \text{(6.36)}$$

for all $k \geqslant 0$ provided that

$$\gamma < \frac{1}{\delta_{s+2\hat{s}} - \delta_{s+\hat{s}} + 1}, \qquad\qquad \text{and} \qquad\qquad (6.37\text{a})$$

$$\gamma(1 - \delta_{s+\hat{s}}) \geqslant 1 - \frac{1}{\sqrt{32}} \approx 0.82, \qquad\qquad (6.37\text{b})$$

Putting these two results together we have

$$\left\| \mathbf{r}^{(i+1)} \right\|_2 \leqslant \frac{1}{2} \left\| \mathbf{r}^{(i)} \right\|_2 + 2\gamma\sqrt{1 + \delta_{s+\hat{s}}} \left\| \mathbf{e} \right\|_2, \qquad\qquad (6.38)$$

if either of the following conditions (a) or (b) are met

(a) $\qquad \gamma \geqslant \dfrac{1}{\delta_{s+2\hat{s}} - \delta_{s+\hat{s}} + 1},\quad$ and $\quad \gamma\delta_{s+2\hat{s}} \leqslant \dfrac{1}{\sqrt{32}},\qquad$ or $\qquad\quad (6.39\text{a})$

(b) $\qquad \gamma < \dfrac{1}{\delta_{s+2\hat{s}} - \delta_{s+\hat{s}} + 1},\quad$ and $\quad \gamma(1 - \delta_{s+\hat{s}}) \geqslant 1 - \dfrac{1}{\sqrt{32}},\qquad\quad (6.39\text{b})$

completing the proof of the lemma. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\square$

Looking at our algorithm in another way shows that the MFR algorithm is capable of attaining the bounds of Lemma 2.7. Lemma 2.7 says that given $\mathbf{y} = \Phi\mathbf{x}$, then the minimiser to $\|\hat{\mathbf{x}}\|_0$ subject to $\mathbf{y} = \Phi\hat{\mathbf{x}}$ is unique and equal to $\mathbf{x}$. Lemma 6.5 and its corollary show that the MFR algorithm can achieve this bound.

Lemma 6.5. *Under the conditions of Proposition 6.2 and for measurements* $\mathbf{y} = \Phi\mathbf{x} + \mathbf{e}$ *where* $\Phi \in \mathbb{R}^{m \times n}$ *has the RIP such that* γ *satisfies*

$$\frac{1 - \frac{\alpha}{2}}{1 - \delta_{s+\hat{s}}} < \gamma < \frac{1}{1 - \delta_{s+\hat{s}}}, \qquad\qquad (6.40)$$

for some constant $0 < \alpha < 1$ *and* $\hat{s} \geqslant s$, *and*

$$\delta_{s+\hat{s}} < 1, \qquad\qquad (6.41)$$

Algorithm 6.1 produces an approximation $\mathbf{x}^{(k)}$ *satisfying*

$$\left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2 \leqslant \alpha^k \|\mathbf{x}\|_2 + \frac{2\gamma\sqrt{1 + \delta_{s+\hat{s}}}}{1 - \alpha} \|\mathbf{e}\|_2. \qquad\qquad (6.42)$$

Proof. Recall the fundamental step of the MFR algorithm

$$\mathbf{x}^{(k+1)} = \mathbb{H}_{\hat{s}} \left(\mathbf{x}^{(k)} + \gamma\Phi^{\mathsf{T}} \left(\mathbf{y} - \Phi\mathbf{x}^{(k)} \right) \right). \qquad\qquad (6.43)$$

As before, set $B^{(k)} \triangleq \operatorname{supp}(\mathbf{x}) \cup \operatorname{supp}(\mathbf{x}^{(k)})$ and $\mathbf{a}^{(k+1)} \triangleq \mathbf{x}^{(k)} + \gamma\Phi^{\mathsf{T}} \left(\mathbf{y} - \Phi\mathbf{x}^{(k)} \right)$. Then we have

$$\mathbf{a}^{(k+1)}_{B^{(k+1)}} = \mathbf{x}^{(k)}_{B^{(k+1)}} + \gamma\Phi^{\mathsf{T}}_{B^{(k+1)}} \left(\mathbf{y} - \Phi\mathbf{x}^{(k)}_{B^{(k+1)}} \right) \qquad\qquad (6.44\text{a})$$

$$= \mathbf{x}^{(k)}_{B^{(k+1)}} + \gamma\Phi^{\mathsf{T}}_{B^{(k+1)}}\Phi(\mathbf{x} - \mathbf{x}^{(k)}) + \gamma\Phi^{\mathsf{T}}_{B^{(k+1)}}\mathbf{e}. \qquad\qquad (6.44\text{b})$$

This gives the error estimate

$$\left\| \mathbf{x} - \mathbf{x}^{(k+1)} \right\|_2 \leqslant \left\| \mathbf{x} - \mathbf{a}_{B^{(k+1)}}^{(k+1)} \right\|_2 + \left\| \mathbf{a}_{B^{(k+1)}}^{(k+1)} - \mathbf{x}^{(k+1)} \right\|_2 \tag{6.45a}$$

$$\leqslant 2 \left\| \mathbf{x} - \mathbf{a}_{B^{(k+1)}}^{(k+1)} \right\|_2 \tag{6.45b}$$

$$= 2 \left\| \mathbf{x} - \mathbf{x}_{B^{(k+1)}}^{(k)} - \gamma \Phi_{B^{(k+1)}}^{\mathsf{T}} \Phi \left(\mathbf{x} - \mathbf{x}_{B^{(k+1)}}^{(k)} \right) - \gamma \Phi_{B^{(k+1)}}^{\mathsf{T}} \mathbf{e} \right\|_2 \tag{6.45c}$$

$$\leqslant 2 \left\| \mathbf{x} - \mathbf{x}_{B^{(k+1)}}^{(k)} - \gamma \Phi_{B^{(k+1)}}^{\mathsf{T}} \Phi \left(\mathbf{x} - \mathbf{x}_{B^{(k+1)}}^{(k)} \right) \right\|_2 + 2 \left\| \gamma \Phi^{\mathsf{T}} \mathbf{e} \right\|_2 \tag{6.45d}$$

$$= 2 \left\| \left(\mathbf{I} - \gamma \Phi^{\mathsf{T}} \Phi \right) \left(\mathbf{x} - \mathbf{x}^{(k)} \right) \right\|_2 + 2\gamma \left\| \Phi_{B^{(k+1)}}^{\mathsf{T}} \mathbf{e} \right\|_2 \tag{6.45e}$$

$$\leqslant 2 \left(1 - \gamma \left(1 - \delta_{s+\hat{s}} \right) \right) \left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2 + 2\gamma \sqrt{1 + \delta_{s+\hat{s}}} \left\| \mathbf{e} \right\|_2 , \tag{6.45f}$$

by Lemmas 3.10 and 3.11. Applying Lemma 6.3 to the recursion defined by Equation (6.45f) we get that

$$\left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2 \leqslant \left[2 \left(1 - \gamma \left(1 - \delta_{s+\hat{s}} \right) \right) \right]^k \left\| \mathbf{x} \right\|_2 + \frac{2\gamma \sqrt{1 + \delta_{s+\hat{s}}}}{1 - 2 \left(1 - \gamma \left(1 - \delta_{s+\hat{s}} \right) \right)} \left\| \mathbf{e} \right\|_2 , \tag{6.46}$$

since $\mathbf{x}^{(0)} = \mathbf{0}$ and if $0 < 2 \left(1 - \gamma \left(1 - \delta_{s+\hat{s}} \right) \right) \leqslant \alpha < 1$. Thus, if

$$0 < 2(1 - \gamma(1 - \delta_{s+\hat{s}})) \leqslant \alpha \iff 1 - \frac{\alpha}{2} \leqslant \gamma(1 - \delta_{s+\hat{s}}) < 1 \tag{6.47a}$$

$$\iff \frac{1 - \frac{\alpha}{2}}{1 - \delta_{s+\hat{s}}} \leqslant \gamma < \frac{1}{1 - \delta_{s+\hat{s}}} , \tag{6.47b}$$

the algorithm will converge provided $\delta_{s+\hat{s}} < 1$ producing an approximation that obeys

$$\left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2 \leqslant \alpha^k \left\| \mathbf{x} \right\|_2 + \frac{2\gamma \sqrt{1 + \delta_{s+\hat{s}}}}{1 - \alpha} \left\| \mathbf{e} \right\|_2 \tag{6.48a}$$

$$\leqslant \alpha^k \left\| \mathbf{x} \right\|_2 + \frac{2\sqrt{2}\gamma}{1 - \alpha} \left\| \mathbf{e} \right\|_2 , \tag{6.48b}$$

completing the lemma. $\square$

This lemma says that for the right value of γ and provided that $\delta_{s+\hat{s}} < 1$, the algorithm will always converge to the correct solution, at the cost of (perhaps quite significant) noise amplification.

Corollary 6.6. *Under the hypothesis of Lemma 6.5 and if we could measure the signal* $\mathbf{y}$ *exactly, i.e. if* $\mathbf{e} = \mathbf{0}$*, then for some value of* γ*, setting* $\hat{s} = s$ *gives an algorithm capable of attaining the bound in Lemma 2.7.*

Proof. Setting $\hat{s} = s$ in Equation (6.41) gives the condition $\delta_{2s} < 1$ and by choosing γ so that

$$\frac{1 - \frac{\alpha}{2}}{1 - \delta_{s+\hat{s}}} \leqslant \gamma < \frac{1}{1 - \delta_{s+\hat{s}}} , \tag{6.49}$$

the algorithm will produce a sequence of approximations which converge to $\mathbf{x}$. $\square$

What is interesting about these results, is that they rely only on the *lower* RIP constant, that is, it

only requires

$$0 < 1 - \delta_{2s} \leqslant \frac{\|\Phi\mathbf{x}\|_2^2}{\|\mathbf{x}\|_2^2}, \tag{6.50}$$

for all $2s$-sparse vectors $\mathbf{x}$. This means that the MFR algorithm will recover the correct s-sparse solution $\mathbf{x}$ to $\mathbf{y} = \Phi\mathbf{x}$ provided that there are no $2s$-sparse vectors in the kernel of Φ. In fact this is a tight theoretical bound, assume that there exists a vector $\mathbf{v} \in \mathbb{R}^n$ that is $2s$-sparse and $\Phi\mathbf{v} = \mathbf{0}$. Choose a set $\Gamma \subset \{1, 2, \ldots, n\}$ of size s so that $\Gamma \subset \mathrm{supp}(\mathbf{v})$ and set $\mathbf{x} = -\mathbf{v}_\Gamma$ so that $\mathbf{x}$ is s-sparse. Then

$$\mathbf{y} = \Phi\mathbf{x} = \Phi\mathbf{x} + \Phi\mathbf{v} = \Phi(\mathbf{x} + \mathbf{v}) = \Phi\mathbf{u}, \tag{6.51}$$

where $\mathbf{u} = \mathbf{x} + \mathbf{v}$ is s-sparse and $\mathbf{u} \neq \mathbf{x}$. Hence there is no unique minimiser to $\|\hat{\mathbf{x}}\|_0$ subject to $\mathbf{y} = \Phi\hat{\mathbf{x}}$ and no algorithm will be able to return the correct solution 100% of the time. Thus the MFR algorithm is able to attain the bound in Lemma 2.7.

Perhaps even more surprising is that the algorithm will converge as fast as we want (but still linearly), i.e. $\left\|\mathbf{x} - \mathbf{x}^{(k)}\right\|_2 \leqslant \alpha^k \|\mathbf{x}\|_2$ for any $0 < \alpha < 1$, provided we can choose γ so that $1 - \frac{\alpha}{2} \leqslant \gamma(1 - \delta_{2s}) < 1$.

This seems to be an astounding result, until we realise that this requires accurate values of δ_{2s} and explicitly calculating δ_{2s} for a random matrix is computationally equivalent to directly solving the ℓ_0 minimisation problem $\arg\min \|\hat{\mathbf{x}}\|_0$ subject to $\mathbf{y} = \Phi\hat{\mathbf{x}}$.

We now use Proposition 6.2 and Lemma 3.8 from (Needell and Tropp 2008) to prove the main theorem.

Proof of Theorem 6.1. We set $\hat{s} = s$, then let $\mathbf{x}^s$ be the best s-sparse approximation to $\mathbf{x}$. Then observe that

$$\left\|\mathbf{x} - \mathbf{x}^{(k)}\right\|_2 \leqslant \|\mathbf{x} - \mathbf{x}^s\|_2 + \left\|\mathbf{x}^{(k)} - \mathbf{x}^s\right\|_2. \tag{6.52}$$

We then apply the algorithm to $\mathbf{y}$ to recover an s-sparse approximation. From Proposition 6.2 we get the bound

$$\left\|\mathbf{x}^{(k)} - \mathbf{x}^s\right\|_2 \leqslant 2^{-k} \|\mathbf{x}^s\|_2 + 4\gamma\sqrt{1 + \delta_{2s}} \|\hat{\mathbf{e}}\|_2, \tag{6.53}$$

where $\hat{\mathbf{e}} = \mathbf{y} - \Phi\mathbf{x}^s$. Using Lemma 3.8 and setting $t = 2s$ we can write $\mathbf{y} = \Phi\mathbf{x}^{2s} + \tilde{\mathbf{e}}$ where $\mathbf{x}^{2s}$ is a best $2s$-sparse approximation to $\mathbf{x}$, such that

$$\|\tilde{\mathbf{e}}\|_2 \leqslant \sqrt{1 + \delta_{2s}} \left(\left\|\mathbf{x} - \mathbf{x}^{2s}\right\|_2 + \frac{1}{\sqrt{2s}} \left\|\mathbf{x} - \mathbf{x}^{2s}\right\|_1 \right) + \|\mathbf{e}\|_2, \tag{6.54}$$

Hence combining Lemma 3.8 and Proposition 6.2 we get

$$\left\|\mathbf{x}^{(k)} - \mathbf{x}^s\right\|_2 \leqslant 2^{-k} \|\mathbf{x}^s\|_2 + 4\gamma\sqrt{1 + \delta_{2s}} \|\mathbf{e}\|_2 + 4\gamma(1 + \delta_{2s}) \left(\left\|\mathbf{x} - \mathbf{x}^{2s}\right\|_2 + \frac{1}{\sqrt{2s}} \left\|\mathbf{x} - \mathbf{x}^{2s}\right\|_1 \right), \tag{6.55}$$

under conditions (a), (b) or (c). This completes the proof of the theorem. $\qquad\square$

Note that in condition (a), namely Equation (6.39a), setting $\gamma = 1/(1 + \delta_s)$ and $\hat{s} = s$ gives the same conditions on convergence for the proof of the IHT algorithm in (Blumensath and Davies 2008). Our result (and algorithm) is a generalisation of this IHT algorithm. We choose to use the RIP rather than the modified-RIP, unlike the authors of (Blumensath and Davies 2008), as it

leads to easier implementation. To implement IHT one requires a measurement matrix scaled by $1/(1 + \delta_s)$, but it is unfeasible to perform this operation exactly. Hence by avoiding this scaling and choosing a deliberately smaller γ (which admittedly disguises some of the difficulty in this scenario) we can much more easily use the unscaled variant.

Observe also that this theorem implies that as $\gamma \to 0$, the error due to the noise component in the model also goes to 0.

Strictly speaking, there is no real reason to require $\left\| \mathbf{r}^{(k+1)} \right\|_2 \leqslant \frac{1}{2} \left\| \mathbf{r}^{(k)} \right\|_2 + 2\gamma\sqrt{1 + \delta_{2s}} \left\| \mathbf{e} \right\|_2$, any value $0 \leqslant \alpha < 1$ with $\left\| \mathbf{r}^{(k+1)} \right\|_2 \leqslant \alpha \left\| \mathbf{r}^{(k)} \right\|_2 + 2\gamma\sqrt{1 + \delta_{2s}} \left\| \mathbf{e} \right\|_2$ would suffice, but perhaps offer significantly slower convergence. What happens to the convergence conditions if we allow a larger α?

Assume we have $\left\| \mathbf{r}^{(k+1)} \right\|_2 \leqslant \alpha \left\| \mathbf{r}^{(k)} \right\|_2 + 2\gamma\sqrt{1 + \delta_{2s}} \left\| \mathbf{e} \right\|_2$, applying Lemma 6.3 to this recursive sequence it follows that

$$\left\| \mathbf{r}^{(k)} \right\|_2 \leqslant \alpha \left\| \mathbf{r}^{(k-1)} \right\|_2 + 2\gamma\sqrt{1 + \delta_{2s}} \left\| \mathbf{e} \right\|_2 \tag{6.56a}$$

$$\leqslant \alpha^k \left\| \mathbf{r}^{(0)} \right\|_2 + 2\gamma\sqrt{1 + \delta_{2s}} \left\| \mathbf{e} \right\|_2 \sum_{i=1}^{k} \alpha^i \tag{6.56b}$$

$$< \alpha^k \left\| \mathbf{r}^{(0)} \right\|_2 + \frac{2\gamma\sqrt{1 + \delta_{2s}}}{1 - \alpha} \left\| \mathbf{e} \right\|_2 \tag{6.56c}$$

$$< \alpha^k \left\| \mathbf{r}^{(0)} \right\|_2 + \frac{2\sqrt{2}\gamma}{1 - \alpha} \left\| \mathbf{e} \right\|_2, \tag{6.56d}$$

which unfortunately threatens significant noise amplification, especially as α gets close to 1. Then the convergence criteria become

(a) $\quad \gamma \geqslant \dfrac{1}{\delta_{s+2\hat{s}} - \delta_{s+\hat{s}} + 1}$, and $\quad \gamma\delta_{s+2\hat{s}} \leqslant \dfrac{\alpha}{\sqrt{8}} < \dfrac{1}{\sqrt{8}} \approx 0.35$, or $\tag{6.57a}$

(b) $\quad \gamma < \dfrac{1}{\delta_{s+2\hat{s}} - \delta_{s+\hat{s}} + 1}$, and $\quad \gamma(1 - \delta_{s+\hat{s}}) \geqslant 1 - \dfrac{\alpha}{\sqrt{8}} > 1 - \dfrac{1}{\sqrt{8}} \approx 0.65, \tag{6.57b}$

which are slightly looser than before.

6.4 Convergence

If Algorithm 6.1 converges, it either converges to the correct sparse solution, or it converges to another sparse vector, but one that is not an (approximate – to some level of tolerance) solution to the equation $\mathbf{y} = \Phi\mathbf{x} + \mathbf{e}$. If the algorithm converges to an incorrect vector, it is simple to test this and, if necessary, rerun the algorithm with different γ or $\hat{s}$ estimates.

Let $\hat{\mathbf{x}} \in \mathbb{R}^n$ be a solution the MFR algorithm converges to. Let $\mathfrak{P} \in \mathbb{R}^{n \times n}$ be a diagonal matrix that is the matrix corresponding to the projection onto the components specified by the support of $\hat{\mathbf{x}}$, i.e. onto the non-zero components of $\hat{\mathbf{x}}$, so $\mathfrak{P}$ is 0 everywhere, except for the diagonal elements $\mathfrak{P}_{ii}$ which are 1 if and only if $\hat{x}_i \neq 0$. By construction we have $\mathfrak{P}^2 = \mathfrak{P} = \mathfrak{P}^{\mathsf{T}}$ and

$\mathfrak{P}\hat{\mathbf{x}} = \hat{\mathbf{x}}$. As the algorithm has converged to $\hat{\mathbf{x}}$, we have

$$\hat{\mathbf{x}} = \mathfrak{P}\left[\hat{\mathbf{x}} + \gamma\Phi^\mathsf{T}\Phi\left(\mathbf{x} - \hat{\mathbf{x}}\right)\right] \Rightarrow \hat{\mathbf{x}} = \hat{\mathbf{x}} + \mathfrak{P}\gamma\Phi^\mathsf{T}\Phi\left(\mathbf{x} - \hat{\mathbf{x}}\right) \tag{6.58a}$$

$$\Rightarrow \mathfrak{P}\Phi^\mathsf{T}\Phi\mathbf{x} = \mathfrak{P}\Phi^\mathsf{T}\Phi\hat{\mathbf{x}} \tag{6.58b}$$

$$\Rightarrow \mathfrak{P}\Phi^\mathsf{T}\Phi\mathbf{x} = \mathfrak{P}^\mathsf{T}\Phi^\mathsf{T}\Phi\mathfrak{P}\hat{\mathbf{x}}. \tag{6.58c}$$

Recall that the Moore-Penrose pseudo-inverse of a matrix A is given by, if it exists, $A^\dagger = (A^*A)^{-1}A^*$. Let $A = \Phi\mathfrak{P}$, then Equation (6.58c) can be rewritten as

$$A^\mathsf{T}\Phi\mathbf{x} = A^\mathsf{T}A\hat{\mathbf{x}}. \tag{6.59}$$

If we can invert $A^\mathsf{T}A$, then $\hat{\mathbf{x}}$ is given by

$$\hat{\mathbf{x}} = (A^\mathsf{T}A)^{-1}A^\mathsf{T}\Phi\mathbf{x} = (\Phi\mathfrak{P})^\dagger\Phi\mathbf{x} = (\Phi\mathfrak{P})^\dagger\mathbf{y}. \tag{6.60}$$

This result is not at all surprising, it merely confirms that if we knew the support of $\mathbf{x}$ (which is encoded in the matrix $\mathfrak{P}$), then we could find the solution by solving the least squares problem restricted to vectors that have this support.

6.5 Adaptive Choice of γ

Recall that the MFR algorithm performs the iterative step

$$\mathbf{x}^{(k+1)} = \mathbb{H}_{\hat{s}}\left[\mathbf{x}^{(k)} + \gamma\Phi^\mathsf{T}\Phi\left(\mathbf{x} - \mathbf{x}^{(k)}\right)\right]. \tag{6.61}$$

Previously we have only considered a fixed value of γ across all iterations, but there is no reason why γ cannot vary from one iteration to the next. We then ask, what is a good way to choose γ at each iteration?

6.5.1 Minimising the Residual

The easiest option is to consider a greedy strategy, so that at every iteration we minimise the ℓ_2-norm of the residual, i.e. $\left\|\mathbf{x} - \mathbf{x}^{(k)}\right\|_2$. So at the iteration producing the output $\mathbf{x}^{(k)}$ we want to choose $\gamma^{(k)}$ so that $\left\|\mathbf{x} - \mathbf{x}^{(k)}\right\|_2$ is minimised, namely

$$\gamma^{(k)} = \arg\min_\gamma \left\|\mathbf{x} - \mathbb{H}_{\hat{s}}\left[\mathbf{x}^{(k-1)} + \gamma\Phi^\mathsf{T}\Phi\left(\mathbf{x} - \mathbf{x}^{(k-1)}\right)\right]\right\|_2. \tag{6.62}$$

Similar to before we rewrite the hard-thresholding function as a projection matrix $\mathfrak{P}$, so that $\mathfrak{P}$ is a diagonal matrix with ones along the diagonal corresponding to the $\hat{s}$ largest components of the update $\mathbf{x}^{(k)} + \gamma\Phi^\mathsf{T}\Phi\left(\mathbf{x} - \mathbf{x}^{(k)}\right)$. Since the position of the largest components can clearly vary from one iteration to the next and for each iteration the position of the largest components is also dependent on $\gamma^{(k)}$ we will write $\mathfrak{P}_\gamma^{(k)}$ to emphasise the dependence on both the iteration and the value of γ for that iteration.

The problem of choosing $\gamma^{(k)}$ in this manner is equivalent to solving

$$\frac{d}{d\gamma} \left\| \mathbf{x} - \mathbf{x}_\gamma^{(k)} \right\|_2^2 = 0, \tag{6.63}$$

for γ where $\mathbf{x}_\gamma^{(k)} \triangleq \mathbb{H}_{\hat{s}} \left[\mathbf{x}^{(k-1)} + \gamma \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi} \left(\mathbf{x} - \mathbf{x}^{(k-1)} \right) \right]$. Using the projection matrix format for $\mathbb{H}_{\hat{s}}$ we get

$$\frac{d}{d\gamma} \left\| \mathbf{x} - \mathbf{x}_\gamma^{(k)} \right\|_2^2 = 0, \tag{6.64a}$$

$$\iff \frac{d}{d\gamma} \left(\left[\mathbf{x} - \mathbf{x}_\gamma^{(k)} \right]^\mathsf{T} \left[\mathbf{x} - \mathbf{x}_\gamma^{(k)} \right] \right) = 0, \tag{6.64b}$$

$$\iff \frac{d}{d\gamma} \left(\left[\mathbf{x} - \mathfrak{P}_\gamma^{(k)} \left(\mathbf{x} - \gamma \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi}(\mathbf{x} - \mathbf{x}^{(k)}) \right) \right]^\mathsf{T} \left[\mathbf{x} - \mathfrak{P}_\gamma^{(k)} \left(\mathbf{x} - \gamma \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi}(\mathbf{x} - \mathbf{x}^{(k)}) \right) \right] \right) = 0 \tag{6.64c}$$

$$\iff \frac{d}{d\gamma} \left(\mathbf{x}^\mathsf{T} \mathbf{x} - \mathbf{x}^\mathsf{T} \mathfrak{P}_\gamma^{(k)} \mathbf{x} - \gamma^{(k)} \mathbf{x}^\mathsf{T} \mathfrak{P}_\gamma^{(k)} \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi}(\mathbf{x} - \mathbf{x}^{(k)}) + \dots \right.$$
$$- \mathbf{x}^{(k)} \mathfrak{P}_\gamma^{(k)} \mathbf{x} + \mathbf{x}^{(k)\mathsf{T}} \mathfrak{P}_\gamma^{(k)} \mathfrak{P}_\gamma^{(k)} \mathbf{x}^{(k)} + \gamma^{(k)} \mathbf{x}^{(k)\mathsf{T}} \mathfrak{P}_\gamma^{(k)} \mathfrak{P}_\gamma^{(k)} \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi}(\mathbf{x} - \mathbf{x}^{(k)}) + \cdots$$
$$- \gamma^{(k)} (\mathbf{x} - \mathbf{x}^{(k)})^\mathsf{T} \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi} \mathfrak{P}_\gamma^{(k)} \mathbf{x} + \gamma^{(k)} (\mathbf{x} - \mathbf{x}^{(k)})^\mathsf{T} \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi} \mathfrak{P}_\gamma^{(k)\mathsf{T}} \mathfrak{P}_\gamma^{(k)} \mathbf{x}^{(k)} + \cdots$$
$$\left. (\gamma^{(k)})^2 (\mathbf{x} - \mathbf{x}^{(k)})^\mathsf{T} \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi} \mathfrak{P}_\gamma^{(k)\mathsf{T}} \mathfrak{P}_\gamma^{(k)} \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi}(\mathbf{x} - \mathbf{x}^{(k)}) \right) = 0 \tag{6.64d}$$

$$\iff -\mathbf{x}^\mathsf{T} \mathfrak{P}_\gamma^{(k)} \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi}(\mathbf{x} - \mathbf{x}^{(k)}) + \mathbf{x}^{(k)\mathsf{T}} \mathfrak{P}_\gamma^{(k)} \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi}(\mathbf{x} - \mathbf{x}^{(k)}) - (\mathbf{x} - \mathbf{x}^{(k)}) \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi} \mathfrak{P}_\gamma^{(k)} \mathbf{x} + \cdots$$
$$+ (\mathbf{x} - \mathbf{x}^{(k)})^\mathsf{T} \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi} \mathfrak{P}_\gamma^{(k)} \mathbf{x}^{(k)} + 2\gamma^{(k)} (\mathbf{x} - \mathbf{x}^{(k)}) \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi} \mathfrak{P}_\gamma^{(k)} \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi}(\mathbf{x} - \mathbf{x}^{(k)}) = 0 \tag{6.64e}$$

$$\iff -(\mathbf{x} - \mathbf{x}^{(k)})^\mathsf{T} \mathfrak{P}_\gamma^{(k)} \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi}(\mathbf{x} - \mathbf{x}^{(k)}) - (\mathbf{x} - \mathbf{x}^{(k)}) \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi} \mathfrak{P}_\gamma^{(k)} (\mathbf{x} - \mathbf{x}^{(k)}) + \cdots$$
$$+ 2\gamma^{(k)} \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi} \mathfrak{P}_\gamma^{(k)} \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi}(\mathbf{x} - \mathbf{x}^{(k)}) = 0 \tag{6.64f}$$

$$\iff \gamma^{(k)} (\mathbf{x} - \mathbf{x}^{(k)})^\mathsf{T} \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi} \mathfrak{P}_\gamma^{(k)} \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi}(\mathbf{x} - \mathbf{x}^{(k)}) = (\mathbf{x} - \mathbf{x}^{(k)})^\mathsf{T} \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi} \mathfrak{P}_\gamma^{(k)} (\mathbf{x} - \mathbf{x}^{(k)}). \tag{6.64g}$$

Hence we have

$$\gamma^{(k)} = \frac{\left(\mathbf{x} - \mathbf{x}^{(k)} \right)^\mathsf{T} \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi} \mathfrak{P}_\gamma^{(k)} \left(\mathbf{x} - \mathbf{x}^{(k)} \right)}{\left(\mathbf{x} - \mathbf{x}^{(k)} \right)^\mathsf{T} \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi} \mathfrak{P}_\gamma^{(k)} \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi} \left(\mathbf{x} - \mathbf{x}^{(k)} \right)} \tag{6.65a}$$

$$= \frac{\mathbf{r}^{(k)\mathsf{T}} \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi} \mathfrak{P}_\gamma^{(k)} \mathbf{r}^{(k)}}{\mathbf{r}^{(k)\mathsf{T}} \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi} \mathfrak{P}_\gamma^{(k)} \boldsymbol{\Phi}^\mathsf{T} \boldsymbol{\Phi} \mathbf{r}^{(k)}}. \tag{6.65b}$$

Thus we can write the optimal step-length in terms of the residual $\mathbf{r}^{(k)} = \mathbf{x} - \mathbf{x}^{(k)}$, the thresholding projection matrix $\mathfrak{P}_\gamma^{(k)}$ and the measurement matrix $\boldsymbol{\Phi}$. Unfortunately however, we cannot calculate the numerator in Equation (6.65b) without knowing the true solution. Unsurprisingly if we implement the MFR algorithm with knowledge of the true solution to calculate the optimal step-length, we achieve a significantly higher reconstruction rate.

What is also interesting to look at, is a graph of $\gamma^{(k)}$ as a function of the iteration count, as in Figure 6.1. Intuitively one might expect the step-length to converge to a constant, and perhaps very small value, but instead, in every simulated case, we see that in the limit, $\gamma^{(k)}$ oscillates between two fixed values. Later work could include further analysis of these values of $\gamma^{(k)}$ to see if there is a good algorithm to estimate these values.

Consider then the MFR algorithm where at every iteration the "optimal" value of γ is chosen in the sense that it minimises the ℓ_2-norm of the residual. We then have the following lemma, Lemma 6.7 regarding its performance.
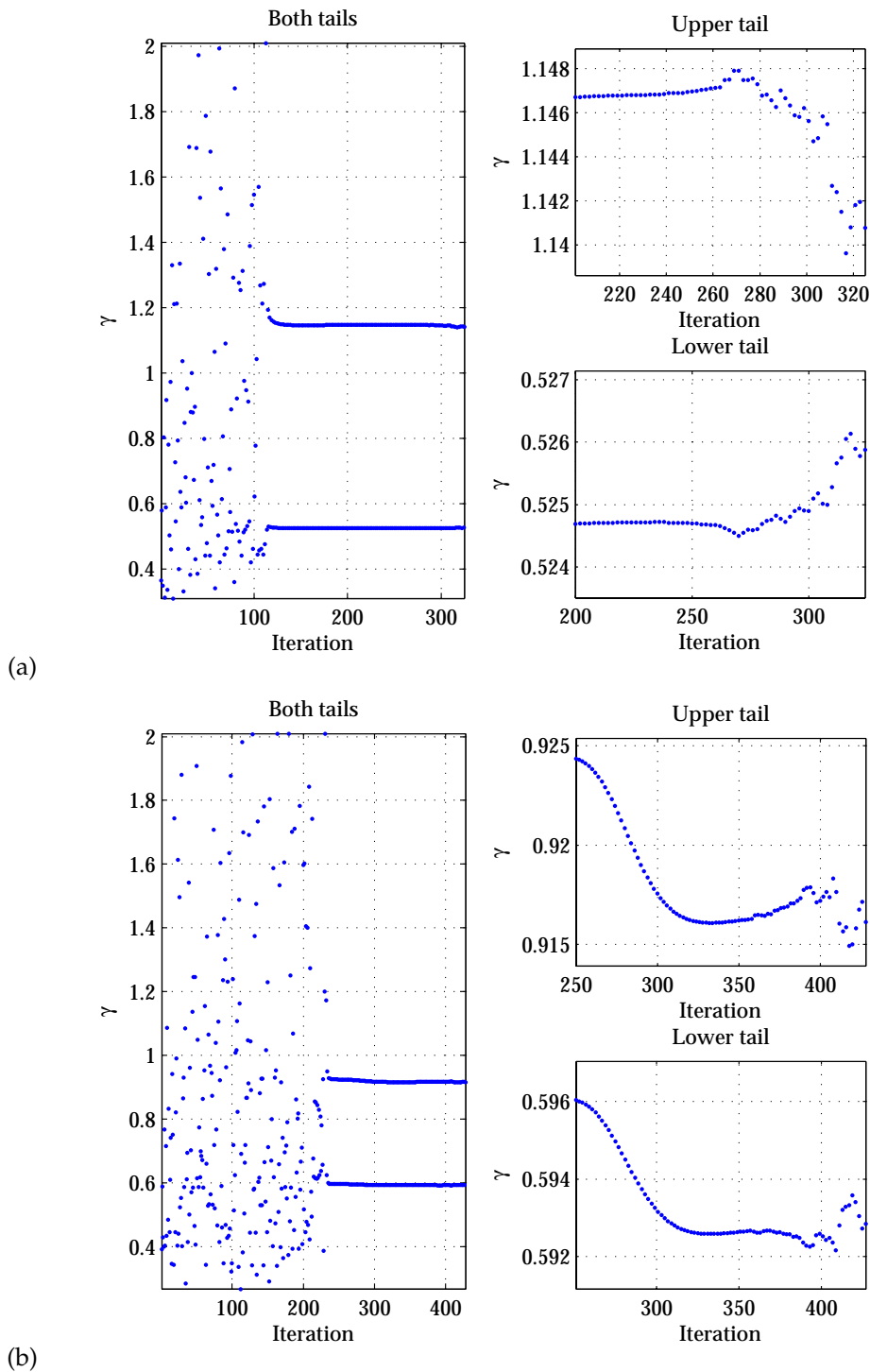
Figure 6.1: Plot of $\gamma^{(k)}$ as a function of the iteration count for two simulations with $n = 400$, $m = 200$ and $s = 100$. Observe how there are two distinct tails for $\gamma^{(k)}$.

Lemma 6.7. *Let $\Phi \in \mathbb{R}^{m \times n}$ be a measurement matrix that obeys the RIP of order s and let $\mathbf{x} \in \mathbb{R}^n$ be the s-sparse signal we are trying to reconstruct given the measurements $\mathbf{y} = \Phi\mathbf{x}$ using the MFR algorithm, Algorithm 6.1 where $\gamma^{(k)}$, the step-length, is chosen at every iteration so as to minimise the*

residual

$$\left\| \mathbf{r}^{(k)} \right\|_2 = \left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2 = \left\| \mathbf{x} - \mathbb{H}_{\hat{s}} \left[\mathbf{x}^{(k-1)} + \gamma \Phi^\mathsf{T} \left(\mathbf{y} - \Phi \mathbf{x}^{(k-1)} \right) \right] \right\|_2. \tag{6.66}$$

Then the ℓ_2-norm of the residual is a non-increasing function of the number of iterations.

Proof. Clearly if $\gamma^{(k)} = 0$ then we have $\mathbf{x}^{(k)} = \mathbf{x}^{(k-1)}$ hence $\left\| \mathbf{r}^{(k)} \right\|_2 = \left\| \mathbf{r}^{(k-1)} \right\|_2$. Thus setting $\gamma^{(k)} = 0$ does not increase the norm of the residual, hence minimising this for $\gamma > 0$ can only further decrease $\left\| \mathbf{r}^{(k)} \right\|_2$. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\square$

6.5.2 Minimising the Residual in Φ-space

As already mentioned, without knowing the true solution $\mathbf{x}$, we cannot calculate the value of $\gamma^{(k)}$ that minimises the residual. As an alternative, we can calculate the value of $\gamma^{(k)}$ that minimises the residual in Φ-space, that is we choose $\gamma^{(k)}$ to minimise the term $\left\| \mathbf{y} - \Phi \mathbf{x}^{(k)} \right\|_2$. Proceeding as before where we write

$$\mathbf{x}^{(k)} = \mathbb{H}_{\hat{s}} \left[\mathbf{x}^{(k-1)} + \gamma^{(k)} \Phi^\mathsf{T} \left(\mathbf{y} - \Phi \mathbf{x}^{(k-1)} \right) \right] = \mathfrak{P}_\gamma^{(k)} \left[\mathbf{x}^{(k-1)} + \Gamma^{(k)} \Phi^\mathsf{T} \left(\mathbf{y} - \Phi \mathbf{x}^{(k-1)} \right) \right], \tag{6.67}$$

and we calculate the derivative of $\mathbf{y} - \Phi \mathbf{x}^{(k+1)}$ with respect to $\gamma^{(k)}$ and solve for 0. First observe that

$$\left\| \mathbf{y} - \Phi \mathbf{x}^{(k+1)} \right\|_2^2 \tag{6.68a}$$

$$= \left(\mathbf{y} - \Phi \mathbf{x}^{(k+1)} \right)^\mathsf{T} \left(\mathbf{y} - \Phi \mathbf{x}^{(k+1)} \right) \tag{6.68b}$$

$$= \mathbf{y}^\mathsf{T} \mathbf{y} - 2 \mathbf{y}^\mathsf{T} \Phi \mathbf{x}^{(k+1)} - \mathbf{x}^{(k+1)\,\mathsf{T}} \Phi^\mathsf{T} \Phi \mathbf{x}^{(k+1)} \tag{6.68c}$$

$$= \mathbf{y}^\mathsf{T} \mathbf{y} - 2 \mathbf{y}^\mathsf{T} \Phi \mathfrak{P}_\gamma^{(k)} \left(\mathbf{x}^{(k)} + \gamma^{(k)} \Phi^\mathsf{T} \Phi (\mathbf{x} - \mathbf{x}^{(k)}) \right) - \cdots$$
$$\qquad - \left(\mathbf{x}^{(k)} + \gamma^{(k)} \Phi^\mathsf{T} \Phi (\mathbf{x} - \mathbf{x}^{(k)}) \right)^\mathsf{T} \mathfrak{P}_\gamma^{(k)} \Phi^\mathsf{T} \Phi \left(\mathbf{x}^{(k)} + \gamma^{(k)} \Phi^\mathsf{T} \Phi (\mathbf{x} - \mathbf{x}^{(k)}) \right) \tag{6.68d}$$

$$= \mathbf{x}^\mathsf{T} \Phi^\mathsf{T} \Phi \mathbf{x} - 2 \mathbf{x}^\mathsf{T} \Phi^\mathsf{T} \Phi \mathfrak{P}_\gamma^{(k)} \mathbf{x}^{(k)} - 2 \gamma^{(k)} \mathbf{x}^\mathsf{T} \Phi^\mathsf{T} \Phi \gamma^{(k)} \Phi^\mathsf{T} \Phi (\mathbf{x} - \mathbf{x}^{(k)}) - \mathbf{x}^{(k)\,\mathsf{T}} \mathfrak{P}_\gamma^{(k)} \Phi^\mathsf{T} \Phi \mathfrak{P}_\gamma^{(k)} \mathbf{x}^{(k)} - \cdots$$
$$\qquad - \gamma^{(k)} \mathbf{x}^{(k)\,\mathsf{T}} \mathfrak{P}_\gamma^{(k)} \Phi^\mathsf{T} \Phi \mathfrak{P}_\gamma^{(k)} \Phi^\mathsf{T} \Phi (\mathbf{x} - \mathbf{x}^{(k)}) - \gamma^{(k)} (\mathbf{x} - \mathbf{x}^{(k)})^\mathsf{T} \Phi^\mathsf{T} \Phi \mathfrak{P}_\gamma^{(k)} \Phi^\mathsf{T} \Phi \mathfrak{P}_\gamma^{(k)} \mathbf{x}^{(k)} - \cdots$$
$$\qquad - \gamma^{(k)\,2} (\mathbf{x} - \mathbf{x}^{(k)})^\mathsf{T} \Phi^\mathsf{T} \Phi \mathfrak{P}_\gamma^{(k)} \Phi^\mathsf{T} \Phi \mathfrak{P}_\gamma^{(k)} \Phi^\mathsf{T} \Phi (\mathbf{x} - \mathbf{x}^{(k)}). \tag{6.68e}$$

Then differentiating with respect to $\gamma^{(k)}$ and solving for 0 we get that

$$2 \mathbf{x}^\mathsf{T} \Phi^\mathsf{T} \Phi \mathfrak{P}_\gamma^{(k)} \Phi^\mathsf{T} \Phi (\mathbf{x} - \mathbf{x}^{(k)}) - \mathbf{x}^{(k)\,\mathsf{T}} \mathfrak{P}_\gamma^{(k)} \Phi^\mathsf{T} \Phi \mathfrak{P}_\gamma^{(k)} \Phi^\mathsf{T} \Phi (\mathbf{x} - \mathbf{x}^{(k)}) - \cdots$$
$$\qquad - (\mathbf{x} - \mathbf{x}^{(k)})^\mathsf{T} \Phi^\mathsf{T} \Phi \mathfrak{P}_\gamma^{(k)} \Phi^\mathsf{T} \Phi \mathfrak{P}_\gamma^{(k)} \mathbf{x}^{(k)} - \cdots$$
$$\qquad - 2 \gamma^{(k)} (\mathbf{x} - \mathbf{x}^{(k)})^\mathsf{T} \Phi^\mathsf{T} \Phi \mathfrak{P}_\gamma^{(k)} \Phi^\mathsf{T} \Phi \mathfrak{P}_\gamma^{(k)} \Phi^\mathsf{T} \Phi (\mathbf{x} - \mathbf{x}^{(k)}) = 0 \tag{6.69a}$$

$$\Longleftrightarrow 2 \mathbf{x}^\mathsf{T} \Phi^\mathsf{T} \Phi \mathfrak{P}_\gamma^{(k)} \Phi^\mathsf{T} \Phi (\mathbf{x} - \mathbf{x}^{(k)}) - 2 \mathbf{x}^{(k)\,\mathsf{T}} \mathfrak{P}_\gamma^{(k)} \Phi^\mathsf{T} \Phi \mathfrak{P}_\gamma^{(k)} \Phi^\mathsf{T} \Phi (\mathbf{x} - \mathbf{x}^{(k)}) - \cdots$$
$$\qquad - 2 \gamma^{(k)} (\mathbf{x} - \mathbf{x}^{(k)})^\mathsf{T} \Phi^\mathsf{T} \Phi \mathfrak{P}_\gamma^{(k)} \Phi^\mathsf{T} \Phi \mathfrak{P}_\gamma^{(k)} \Phi^\mathsf{T} \Phi (\mathbf{x} - \mathbf{x}^{(k)}) = 0 \tag{6.69b}$$

Hence

$$\gamma^{(k)} = \frac{\left(\mathbf{x} - \mathbf{x}^{(k)} \right) \Phi^\mathsf{T} \Phi \mathfrak{P}_\gamma^{(k)} \Phi^\mathsf{T} \Phi \left(\mathbf{x} - \mathbf{x}^{(k)} \right)}{(\mathbf{x} - \mathbf{x}^{(k)})^\mathsf{T} \Phi^\mathsf{T} \Phi \mathfrak{P}_\gamma^{(k)} \Phi^\mathsf{T} \Phi \mathfrak{P}_\gamma^{(k)} \Phi^\mathsf{T} \Phi \left(\mathbf{x} - \mathbf{x}^{(k)} \right)}, \tag{6.70}$$

is the value for $\gamma^{(k)}$ which minimises $\left\| \mathbf{y} - \Phi \mathbf{x}^{(k)} \right\|_2$. In Figures 7.1-7.2 we graph the performance of the decoding algorithm where we use this rule to select the value of γ used in each iteration.

Similar to lemma 6.7, we can easily show that using Equation (6.70) to select the value of γ, the ℓ_2-distance between the measurements $\mathbf{y}$ and $\Phi \mathbf{x}^{(k)}$ never increases.

Lemma 6.8. *Let $\Phi \in \mathbb{R}^{m \times n}$ be a measurement matrix that obeys the RIP of order s and let $\mathbf{x} \in \mathbb{R}^n$ be the s-sparse signal we are trying to reconstruct given the measurements $\mathbf{y} = \Phi \mathbf{x}$ using the MFR algorithm, Algorithm 6.1 where $\gamma^{(k)}$, the step-length, is chosen at every iteration so as to minimise the residual in Φ-space*

$$\left\| \Phi \mathbf{r}^{(k)} \right\|_2 = \left\| \mathbf{y} - \Phi \mathbf{x}^{(k)} \right\|_2 = \left\| \mathbf{y} - \Phi \mathbb{H}_{\hat{s}} \left[\mathbf{x}^{(k-1)} + \gamma \Phi^{\mathsf{T}} \left(\mathbf{y} - \Phi \mathbf{x}^{(k-1)} \right) \right] \right\|_2. \qquad (6.71)$$

Then the ℓ_2-norm of the residual is a non-increasing function of the number of iterations.

Proof. Clearly if $\gamma^{(k)} = 0$ then we have $\mathbf{x}^{(k)} = \mathbf{x}^{(k-1)}$ hence $\left\| \Phi \mathbf{r}^{(k)} \right\|_2 = \left\| \Phi \mathbf{r}^{(k-1)} \right\|_2$. Thus setting $\gamma^{(k)} = 0$ does not increase the norm of the residual, hence minimising this for $\gamma > 0$ can only further decrease $\left\| \Phi \mathbf{r}^{(k)} \right\|_2$. $\qquad \square$

One of the problems though with either of the rules for γ given by Equations (6.65b) or (6.70) is that the projection matrix $\mathfrak{P}_\gamma^{(k)}$ depends on the value of $\gamma^{(k)}$. We suggest two alternatives to get around this problem.

▶ **Method 1:** Use an estimate $\hat{\gamma}$ to calculate $\mathbf{x}^{(k+1)}$ from $\mathbf{x}^{(k)}$ which gives a projection matrix $\mathfrak{P}_{\hat{\gamma}}$. Using this projection matrix, calculate the optimal value of γ using Equations (6.65b) or (6.70). This results in a new estimate for γ.

Repeat this process until the value of γ calculated from Equations (6.65b) or (6.70) returns the same projection matrix. That is, until using the value $\gamma^{(k)}$ to get $\mathbf{x}^{(k+1)}$ gives the projection matrix $\mathfrak{P}_\gamma^{(k)}$ and using Equations (6.65b) or (6.70) to get γ from $\mathfrak{P}$ has $\gamma = \gamma^{(k)}$.

▶ **Method 2:** Use standard minimisation techniques to minimise the non-linear equations

$$\left\| \mathbf{x} - \mathbf{x}^{(k+1)} \right\|_2 \qquad \text{or} \qquad \left\| \mathbf{y} - \Phi \mathbf{x}^{(k+1)} \right\|_2, \qquad (6.72)$$

depending on what we are optimising for.

For completeness we list the generic algorithm using a locally optimal value of γ as Algorithm 6.2.

Finally we can apply the main theorem of this thesis, Theorem 6.1 to our modified algorithm.

Corollary 6.9. *Fix $\hat{s}$ the sparsity of the desired solution. For measurements $\mathbf{y} = \Phi \mathbf{x} + \mathbf{e}$ where $\Phi \in \mathbb{R}^{m \times n}$ has the RIP such that either condition (a), (b) or (c) is satisfied at every iteration*

$$(a) \qquad \gamma^{(k)} \geqslant \frac{1}{\delta_{3s} - \delta_{2s} + 1}, \quad and \quad \gamma^{(k)} \delta_{3s} \leqslant \frac{1}{\sqrt{32}}, \qquad or \qquad (6.73a)$$

$$(b) \qquad \gamma^{(k)} < \frac{1}{\delta_{3s} - \delta_{2s} + 1}, \quad and \quad \gamma^{(k)} (1 - \delta_{2s}) \geqslant 1 - \frac{1}{\sqrt{32}}, \qquad or \qquad (6.73b)$$

$$(c) \qquad \frac{3}{4(1 - \delta_{2s})} < \gamma^{(k)} < \frac{1}{1 - \delta_{2s}}, \quad and \quad \delta_{2s} < 1, \qquad (6.73c)$$

Algorithm 6.2 MFR Algorithm with locally optimal γ

Input:

- ▶ Frame vectors $\{e_i\}$ which are the rows of Φ.
- ▶ Observation vector $\mathbf{y}$.
- ▶ Estimate of sparsity $\hat{s}$ of the vector $\mathbf{x}$.

Output:

- ▶ A solution $\hat{\mathbf{x}}$ to $\mathbf{y} = \Phi\mathbf{x}$.

1: $\mathbf{x}^{(0)} \leftarrow 0$
2: **for** $k = 0, \ldots, \infty$ **do**
3: $\quad \gamma^{(k)} \leftarrow \arg\min_\gamma \left\| \mathbf{y} - \Phi\mathbb{H}_{\hat{s}}\left(\mathbf{x}^{(k)} + \gamma\Phi^\mathsf{T}\left(\mathbf{y} - \Phi\mathbf{x}^{(k)}\right)\right) \right\|_2$
4: $\quad \mathbf{x}^{(k+1)} \leftarrow \mathbb{H}_{\hat{s}}\left(\mathbf{x}^{(k)} + \gamma\Phi^\mathsf{T}\left(\mathbf{y} - \Phi\mathbf{x}^{(k)}\right)\right)$
5: **end for**
6: **return** $\mathbf{x}^{(k)}$

Algorithm 6.1 recovers an approximation $\mathbf{x}^{(k)}$ satisfying

$$\left\|\mathbf{x}^{(k)} - \mathbf{x}^s\right\|_2 \leqslant 2^{-k}\left\|\mathbf{x}^s\right\|_2 + 4\gamma_{\max}\sqrt{1+\delta_{2s}}\left\|\mathbf{e}\right\|_2 + 4\gamma_{\max}(1+\delta_{2s})\left(\left\|\mathbf{x} - \mathbf{x}^{2s}\right\|_2 + \frac{1}{\sqrt{2s}}\left\|\mathbf{x} - \mathbf{x}^{2s}\right\|_1\right),$$
$$\tag{6.74}$$

where $\mathbf{x}^s$ is the best s-sparse approximation to $\mathbf{x}$ and $\gamma_{\max} = \max_k \gamma^{(k)}$, is the maximum value of $\gamma^{(k)}$ used in the algorithm.

Proof. The corollary follows immediately from the theorem since there is no requirement on the $\gamma^{(k)}$ to be constant in the proof of the theorem. $\qquad\square$

Although we have the same convergence bounds in both the theorem and the corollary for different algorithms, since we are choosing an "optimal" value of γ in the second algorithm, we would in fact expect convergence to be faster. In Figure 6.2 we compare the number of iterations required for convergence between the vanilla algorithm and the one with locally optimal γ. This shows that the MFR algorithm with an adaptive step-length converges significantly faster than without.

In Chapter 7 in Figures 7.1-7.2 we show the simulated reconstruction for this algorithm and compare it to various other algorithms.

6.5.3 Analysis of Optimal γ

Although we cannot directly calculate the optimal value of γ at each iteration so as to minimise the future residual, perhaps there is some other way to infer this value. The point of the calculations in Equations (6.64a)-(6.65b) is to see if this quantity is estimable without knowing the residual, unfortunately this does not appear to be the case. In this section we will take a look at the optimal values of γ, calculated in this sense, to see if we can infer a pattern to them.

For all simulated cases we see that as the calculated solution $\mathbf{x}^{(k)}$ approaches the true solution $\mathbf{x}$, the "optimal" value of γ oscillates between two values, as in Figure 6.1. In Figure 6.3 we
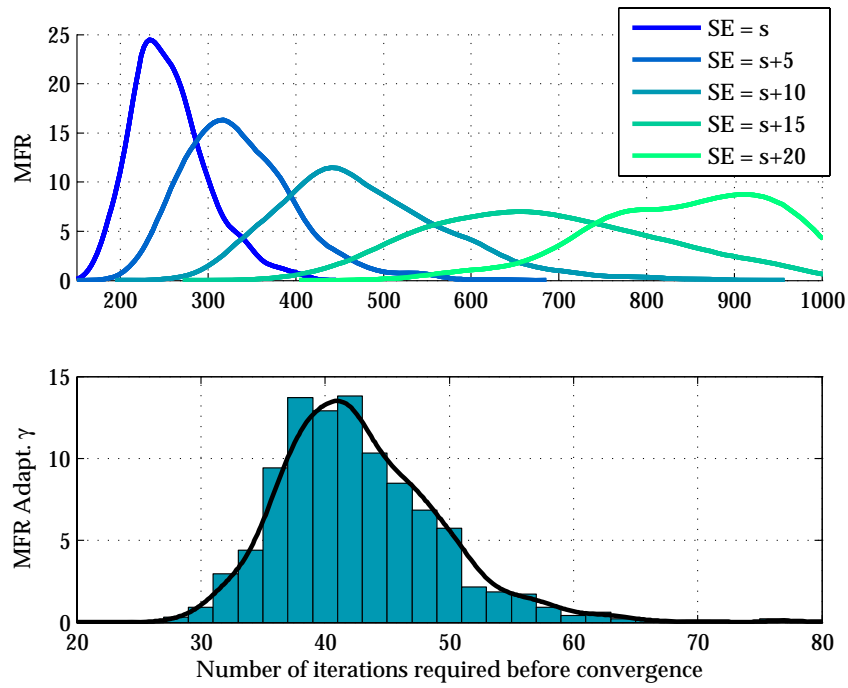
Figure 6.2: Histogram of the number of iterations required for convergence for the MFR algorithm with a constant step-length (top) and using an adaptive step-length (bottom). The *y*-axis shows the relative number of simulations that converged for that bin-width. In the top graph we plot the number of iterations required for various sparsity estimates ranging from $\hat{s} = s$ to $\hat{s} = s + 20$.

produce a scatter plot of the upper tail versus the lower tail for three different matrix sizes. There is a clear clustering of the points for each of the matrix sizes and sparsity levels.

It is also informative to look at a histogram of the distribution of points in the upper and lower tails, which can be seen in Figure 6.4.
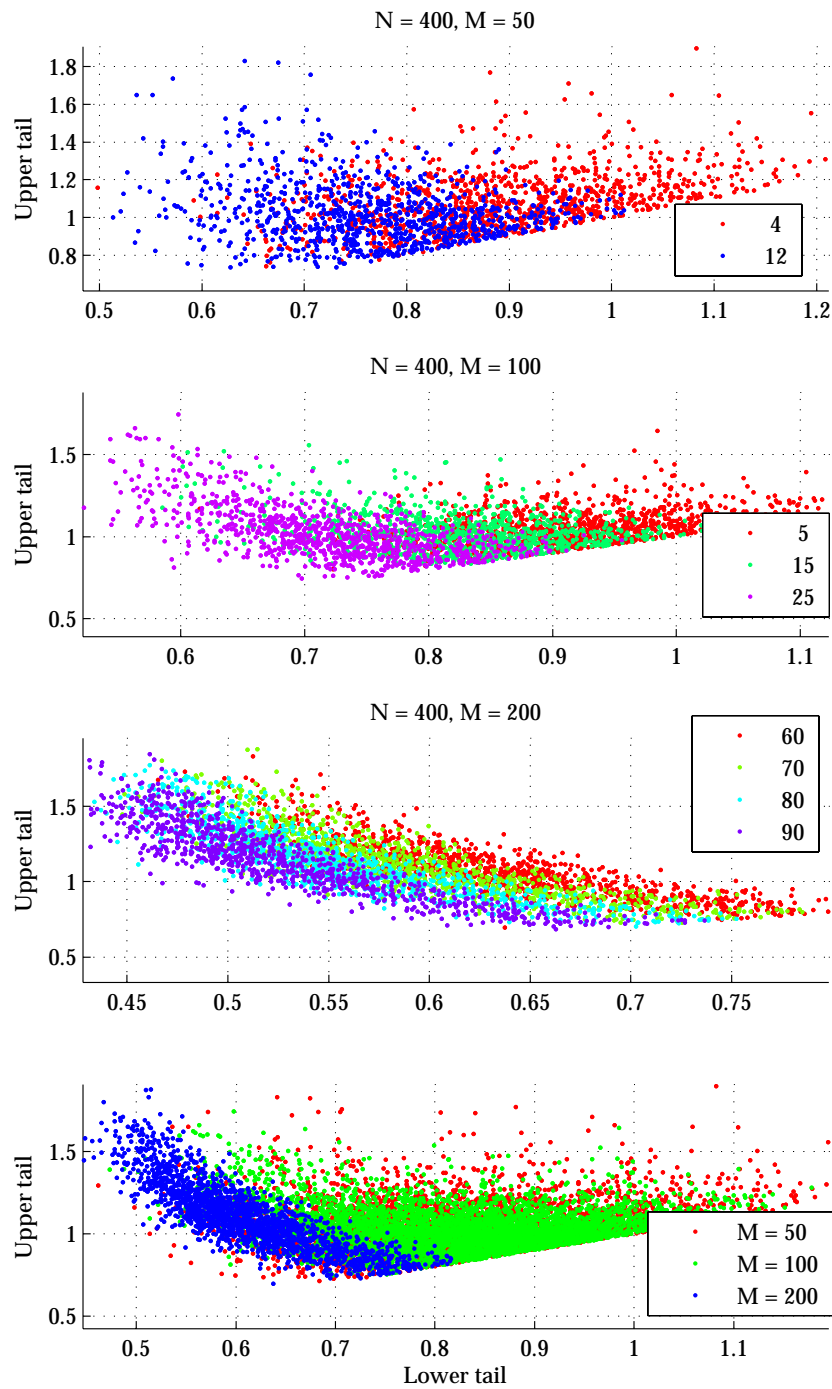
Figure 6.3: Scatter plots of the upper tail versus the lower tail for three different matrix sizes $n = 400$ and $m = 50, 100$ and $m = 200$ (top three plots). For each of these matrix sizes we plot the upper versus lower tail for different sparsity levels. The bottom plot is the overlay of the top three plots without the additional sparsity levels highlighted.
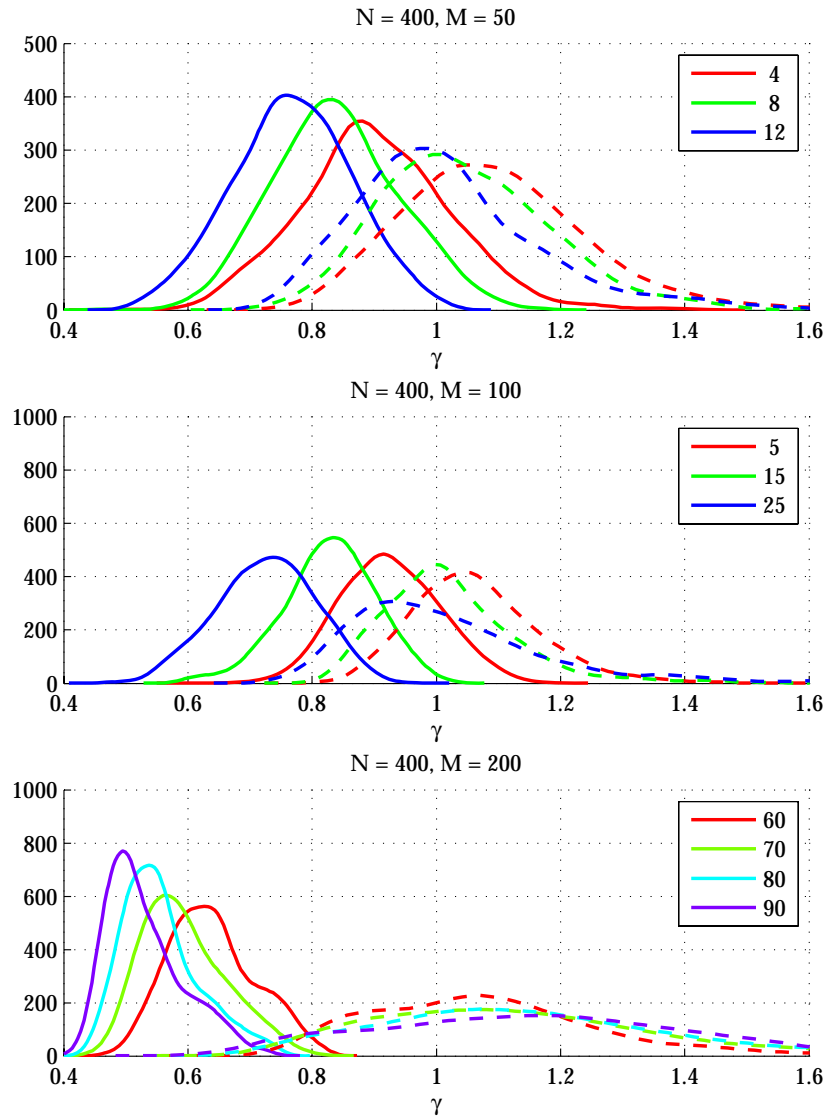
Figure 6.4: Estimated distribution of points in the upper and lower tails.

6.6   Frame Reconstruction with Least Squares

In this section we will demonstrate a modification to the MFR algorithm by stealing a leaf from the orthogonal matching pursuit class of algorithms. The basic idea behind orthogonal matching pursuit algorithms is to:

(a) perform a column selection, and then

(b) solve the problem via least squares minimisation on these columns.

We can then modify our previous frame reconstruction algorithm to also incorporate least squares. We use the frame reconstruction algorithm to provide column selection, by choos-

---

Algorithm 6.3 MFR with Least Squares

Input:

- ▶ The measurement matrix Φ.
- ▶ Observation vector $\mathbf{y}$
- ▶ Estimate of sparsity $\hat{s}$ of the vector $\mathbf{x}$.
- ▶ Step size γ.

Output:

- ▶ A solution $\hat{\mathbf{x}}$ to $\mathbf{y} = \Phi\mathbf{x}$.

1: $\mathbf{x}^{(0)} \leftarrow 0$
2: $\Gamma^{(0)} \leftarrow \varnothing$
3: $\Gamma \leftarrow \{1, \ldots, n\}$
4: **for** $k = 0, \ldots, \infty$ **do**
5:      $\hat{\mathbf{x}} \leftarrow \mathbf{x}^{(k)}$
6:      **while** $\Gamma \neq \Gamma^{(0)}$ **do**   ▷ Loop until support changes
7:          $\hat{\mathbf{x}} \leftarrow \mathbf{x}^{(k)} + \gamma\Phi^{\mathsf{T}}\left(\mathbf{y} - \Phi\hat{\mathbf{x}}\right)$
8:          $\Gamma \leftarrow \operatorname{supp}(\hat{\mathbf{x}}^{\hat{s}})$
9:      **end while**
10:      $\mathbf{x}^{(k+1)} \leftarrow \arg\min_{\tilde{\mathbf{x}}:\ \operatorname{supp}(\tilde{\mathbf{x}})=\Gamma} \|\mathbf{y} - \Phi\tilde{\mathbf{x}}\|_2$ ▷ Solve LS on Γ
11:      $\Gamma^{(0)} \leftarrow \Gamma$
12:      $\Gamma \leftarrow \varnothing$
13: **end for**
14: **return** $\mathbf{x}^{(k+1)}$

---

ing the columns of Φ that correspond to the entries of $\mathbf{x}^{(k)}$ that have largest magnitude and we then solve the least squares problem on this set of columns.

Obviously given the same set of columns twice, to solve the ℓ_2 minimisation problem on, the algorithm will return the same solution both times. Hence our idea is to keep running the frame step until the set of columns we select changes, and then perform least squares, as in Algorithm 6.3. Figures 7.1-7.2 illustrate the performance of the MFR algorithm incorporating a least squares step.

Convergence of this new algorithm is significantly quicker than the plain MFR algorithm. In Figure 6.5 we compare the number of iterations required for convergence for these two algorithms. This shows that the MFR algorithm with a least squares step converges significantly faster than without, indeed it converges about two orders of magnitude faster.

We then have the following lemma about MFR with least squares.

Lemma 6.10. *Let* $\mathbf{x} \in \mathbb{R}^n$ *be an s-sparse signal and let* $\Phi \in \mathbb{R}^{m \times n}$ *be a measurement matrix with RIP constant* δ_{2s} *such that* $\delta_{2s} < 1$. *Set* $\gamma = 2 \cdot \left(\lambda_{\min}(\Phi^{\mathsf{T}}\Phi) + \lambda_{\max}(\Phi^{\mathsf{T}}\Phi)\right)^{-1}$, *then given the measurements* $\mathbf{y} = \Phi\mathbf{x}$, *the MFR algorithm with least squares recovers an approximation* $\mathbf{x}^{(k)}$ *at every iteration satisfying*

$$\left\|\mathbf{x} - \mathbf{x}^{(k)}\right\|_2 \leqslant 2^k \left(\frac{1 + \delta_{2s}}{1 - \delta_{2s}}\right)^{k/2} \left(\frac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}}\right)^{\alpha} \left\|\mathbf{x} - \mathbf{x}^{(0)}\right\|_2, \tag{6.75}$$

where k is the number of times we perform the ℓ_2 *minimisation and* α *is the number of times the frame operation, line 7, is repeated in Algorithm 6.3.*

Furthermore, as soon as the algorithm solves the ℓ_2 problem on support of the true solution, the algorithm will terminate with the correct solution $\mathbf{x}^{(k)} = \mathbf{x}$.

Proof. Assume that on the k-th iteration the algorithm has produced an approximation $\mathbf{x}^{(k)}$ with support $\Gamma^{(k)}$. Set $\mathbf{v}^{(0)} = \mathbf{x}^{(k)}$ and then the algorithm will repeat the frame step $\mathbf{v}^{(i+1)} = \mathbf{v}^{(i)} + \gamma \Phi^\mathsf{T} \left(\mathbf{y} - \Phi \mathbf{v}^{(i)} \right)$ until the support changes. Assume that this takes $\alpha^{(k)}$ iterations before this happens. This is identical to our analysis of the original frame reconstruction algorithm. Let $\lambda_{\min}$ and $\lambda_{\max}$ be the minimum and maximum eigenvalues of $\Phi^\mathsf{T}\Phi$. Then approximation $\mathbf{v}^{\left(\alpha^{(k)}\right)}$ obeys

$$\left\| \mathbf{x} - \mathbf{v}^{\left(\alpha^{(k)}\right)} \right\|_2 \leqslant \left(\frac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}} \right)^{\alpha^{(k)}} \left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2. \tag{6.76}$$

We use our analysis of the frame algorithm, rather than the method used in Theorem 6.1 since the sequence $\mathbf{v}^{(i)}$ are not necessarily sparse.

Now set $\hat{\mathbf{x}} = \mathbb{H}_s(\mathbf{v}^{\left(\alpha^{(k)}\right)})$ and using the fact that $\hat{\mathbf{x}}$ is the best approximation to $\mathbf{v}^{\left(\alpha^{(k)}\right)}$ we see that

$$\left\| \mathbf{x} - \hat{\mathbf{x}} \right\|_2 \leqslant \left\| \mathbf{x} - \mathbf{v}^{\left(\alpha^{(k)}\right)} \right\|_2 + \left\| \mathbf{v}^{\left(\alpha^{(k)}\right)} - \hat{\mathbf{x}} \right\|_2 \tag{6.77a}$$

$$\leqslant 2 \left\| \mathbf{x} - \mathbf{v}^{\left(\alpha^{(k)}\right)} \right\|_2. \tag{6.77b}$$

Now let $\mathbf{x}^{(k+1)}$ be the solution to the problem

$$\arg\min_{\tilde{\mathbf{x}}} \left\| \mathbf{y} - \Phi \tilde{\mathbf{x}} \right\|_2 \quad \text{subject to} \quad \mathrm{supp}(\tilde{\mathbf{x}}) = \mathrm{supp}(\hat{\mathbf{x}}), \tag{6.78}$$

hence

$$\left\| \mathbf{y} - \Phi \mathbf{x}^{(k+1)} \right\|_2 \leqslant \left\| \mathbf{y} - \Phi \hat{\mathbf{x}} \right\|_2 \leqslant \|\Phi\| \cdot \|\mathbf{x} - \hat{\mathbf{x}}\|_2 \leqslant \sqrt{1 + \delta_{2s}} \, \|\mathbf{x} - \hat{\mathbf{x}}\|_2, \tag{6.79}$$

as $\mathbf{x} - \hat{\mathbf{x}}$ has at most $2s$ non-zero components. Also by the RIP we have

$$\left\| \mathbf{y} - \Phi \mathbf{x}^{(k+1)} \right\|_2 \geqslant \sqrt{1 - \delta_{2s}} \, \left\| \mathbf{x} - \mathbf{x}^{(k+1)} \right\|_2. \tag{6.80}$$

Putting all this together, we get

$$\left\| \mathbf{x} - \mathbf{x}^{(k+1)} \right\|_2 \leqslant \sqrt{\frac{1 + \delta_{2s}}{1 - \delta_{2s}}} \, \|\mathbf{x} - \hat{\mathbf{x}}\|_2 \tag{6.81a}$$

$$\leqslant \sqrt{\frac{1 + \delta_{2s}}{1 - \delta_{2s}}} \cdot 2 \left\| \mathbf{x} - \mathbf{v}^{\left(\alpha^{(k)}\right)} \right\|_2 \tag{6.81b}$$

$$\leqslant \sqrt{\frac{1 + \delta_{2s}}{1 - \delta_{2s}}} \cdot 2 \cdot \left(\frac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}} \right)^{\alpha^{(k)}} \left\| \mathbf{x} - \mathbf{x}^{(k)} \right\|_2. \tag{6.81c}$$

Set $\alpha = \sum_k \alpha^{(k)}$ and we get that

$$\left\| \mathbf{x} - \mathbf{x}^{(k+1)} \right\|_2 \leqslant 2^k \left(\frac{1 + \delta_{2s}}{1 - \delta_{2s}} \right)^{k/2} \left(\frac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}} \right)^{\alpha} \left\| \mathbf{x} - \mathbf{x}^{(0)} \right\|_2, \tag{6.82}$$

as required.

As soon as the algorithm correctly identifies the support of $\mathbf{x}$, least squares minimisation will returns an s-sparse vector $\mathbf{x}'$ such that $\|\mathbf{y} - \Phi \mathbf{x}'\|_2 = 0$. As $\delta_{2s} < 1$, there is a unique such

vector and hence $\mathbf{x}' = \mathbf{x}$, the desired solution. The algorithm will then terminate, as the correct solution is a stable point of the frame reconstruction equation and hence convergence will have occurred. □
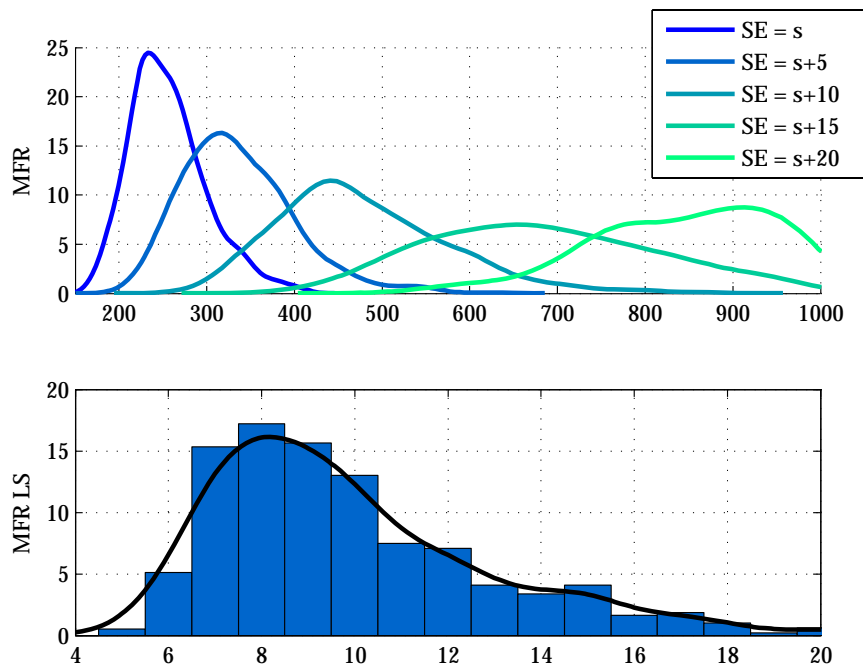
Figure 6.5: Histogram of the number of iterations required for convergence for the plain MFR algorithm (top) and the algorithm with a least squares step (bottom). For the MFR LS algorithm, this is the number of times the frame step is run, i.e. line 7 in Algorithm 6.3. The y-axis shows the relative number of simulations that converged for that bin-width. In the top graph we plot the number of iterations required for various sparsity estimates ranging from $\hat{s} = s$ to $\hat{s} = s + 20$.

6.7 Empirical Results

6.7.1 Reconstruction Rate

We include simulations of our algorithms in Chapter 7, Figures 7.1-7.2.

6.7.2 Convergence Rate

Although we have shown analytically the performance of our three algorithms is similar, the two modified versions both exhibit a significantly faster rate of convergence. In Figure 6.6 we plot histograms of the number of iterations required for convergence for measurement matrices of two different sizes.

Figure 6.7 demonstrates how the rate of convergence tends to increase with the step-length, γ. The following figure, Figure 6.8 also demonstrates how the rate of convergence increases as γ decreases. What is interesting is that convergence appears not to be a smooth function of γ, for

some very small regions of γ the algorithm does converge but then for slightly larger values, does not.
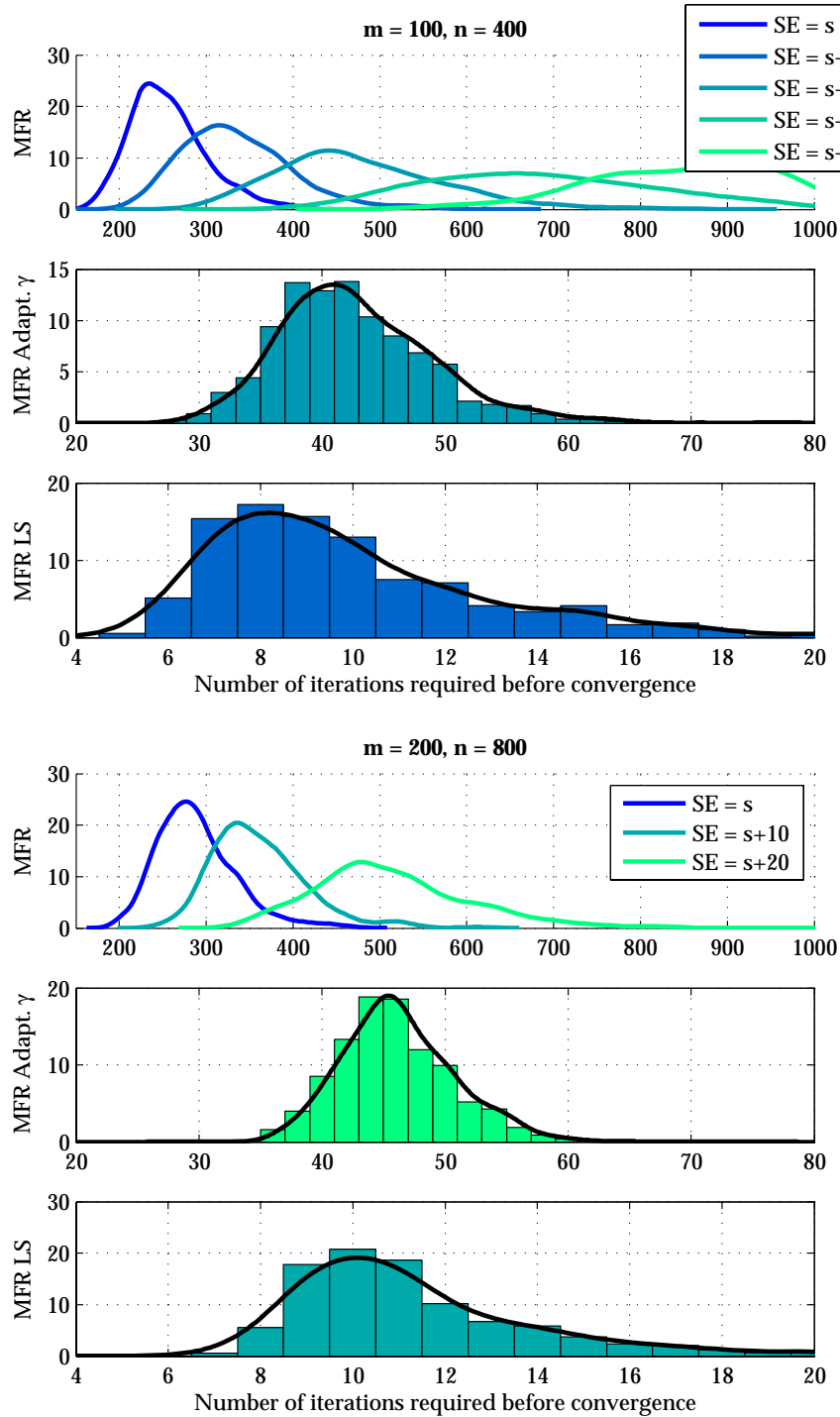
Figure 6.6: Histogram of the number of iterations required for convergence for the MFR algorithm for measurement matrix sizes 100×400 and 200×800 for the following algorithms: MFR (top), MFR with an adaptive step-length (middle) and MFR with least squares (bottom). For the MFR LS algorithm, this is the number of times the frame step is run, i.e. line 7 in Algorithm 6.3. The y-axis shows the relative number of simulations that converged for that bin-width. In the top graph we plot the number of iterations required for various sparsity estimates ranging from $\hat{s} = s$ to $\hat{s} = s + 20$.
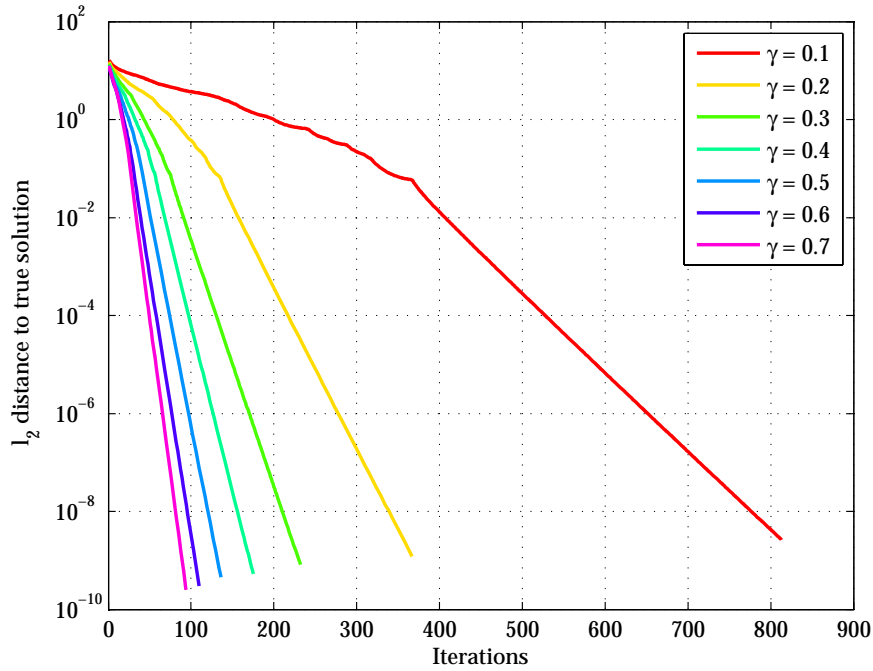
Figure 6.7: Convergence rate for typical scenario. Observe that the larger γ is, the faster the convergence, as we would expect. Convergence is also clearly linear.

6.8   Comments

The various versions of the MFR algorithm we have discussed here have several elements in common with previous algorithms, in particular the Iterative Hard Thresholding algorithm, Algorithm 5.5 (Blumensath and Davies 2008), Gradient Pursuit in Algorithm 5.3 (Blumensath and Davies 2008) and CoSaMP, Algorithm 5.4 (Needell and Tropp 2008), so it is important to note the differences between these algorithms.

6.8.1   Comparison to IHT

The algorithm we have presented generalises the previous IHT algorithm. We need to be careful in directly comparing the two algorithms as they use slightly different measurement matrix structures. The IHT algorithm assumes a scaled matrix $\hat{\Phi}$, that is, let $\Phi \in \mathbb{R}^{m \times n}$ have RIP constant δ_s of order s, then the algorithm reconstructs $\mathbf{x}$ given the measurements

$$\mathbf{y} = \hat{\Phi}\mathbf{x} = \frac{\Phi}{1 + \delta_s}\mathbf{x}, \tag{6.83}$$

whereas we operate directly with the original matrix Φ. This means that by setting $\gamma = \frac{1}{1+\delta_s}$ and $\hat{s} = s$ in our algorithm, the two algorithms are equivalent.

The advantage of allowing γ to to be variable is phenomenal. We have already shown that by allowing γ to be smaller we can dramatically improve the reconstruction rate of the algorithm with low sparsity inputs. Alternatively with very sparse inputs we can increase the convergence rate by choosing the value of γ to be larger. Theorem 6.1 says that, under the hypothesis of the
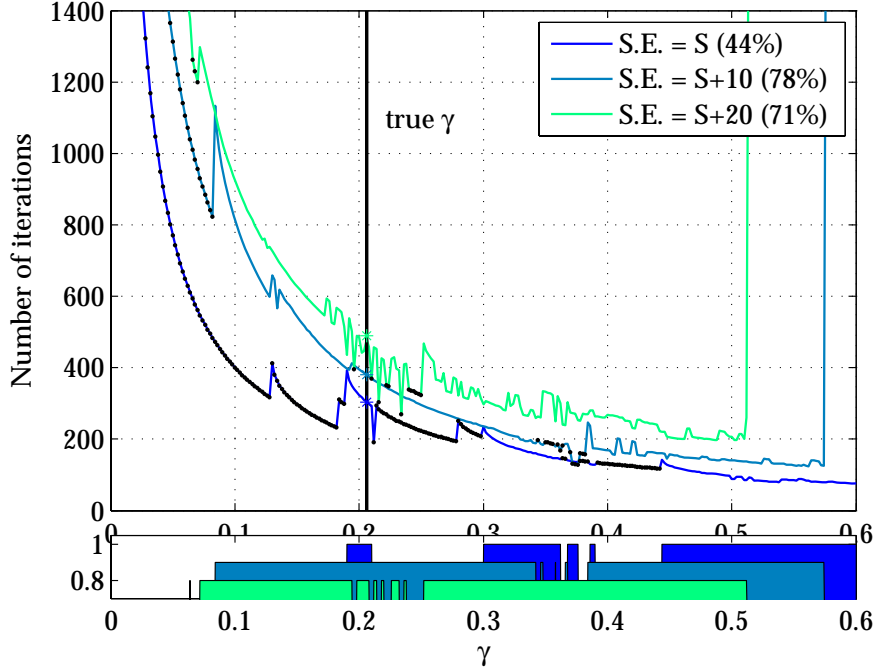
Figure 6.8: Number of iterations required for convergence as a function of γ. Plots shows 3 different sparsity estimates S.E. equal to $s, s + 10$ and $s + 20$. The number in brackets is the fraction of the γ values that it successfully converges for. The plot at the bottom shows the regions in which the algorithm converges and the black dots mark the points where successful convergence does not occur. One would intuitively expect successful convergence to be an almost "continuous" function, but the fact that convergence sometimes occurs in very narrow blocks is strange.

theorem, the error in the final signal is bounded by the term $4\gamma \|\mathbf{e}\|_2$ where $\mathbf{e}$ is the error in measuring the signal. Hence by taking a small value for γ we can decrease the effect of the error in the final output.

In comparison to the analysis of IHT, Theorem 6.1 offers a slightly better convergence result. Setting $\hat{s} = s$ and $\gamma = \frac{1}{1+\delta_s}$ yields the identical condition to the main Theorem of (Blumensath and Davies 2008). But our analysis says that provided $\gamma \geqslant 1/\left(\delta_{s+2\hat{s}} - \delta_{s+\hat{s}} + 1\right)$, if $\delta_{s+2\hat{s}} > 1/\sqrt{32}$ then we can decrease γ and still get convergence. Alternatively, if $\delta_{s+2\hat{s}} < 1/\sqrt{32}$ then we can increase γ to get faster convergence, provided we could estimate the quantities $\delta_{s+\hat{s}}$ and $\delta_{s+2\hat{s}}$ accurately. This property is illustrated in Figure 6.6 where we show how the algorithm converges much quicker for a larger value of γ.

Unfortunately the problem lies in knowing what value of γ to take, so that it falls in the desired range, which requires knowing the RIP constants for a matrix. The only way to do this currently, is to check all $\binom{n}{\tilde{s}}$ submatrices ($\tilde{s} = s + \hat{s}$ or $\tilde{s} = s + 2\hat{s}$ as appropriate) with $\tilde{s}$ columns of Φ, which is computationally unfeasible and is in fact computationally equivalent to directly solving the original ℓ_0 problem, $\arg\min \|\hat{\mathbf{x}}\|_0$ subject to $\mathbf{y} = \Phi\hat{\mathbf{x}}$, directly. There are however fast ways to establish lower bounds on the β and δ parameters of a matrix which we discuss in Chapter 8.

We will discuss this further in Chapter 7, but by requiring a scaled measurement matrix, the implementation of the IHT algorithm becomes complicated. There is no known way to generate measurement matrices with particular RIP constants, although we can say with high probability

they are bounded by a certain value, but to implement the IHT algorithm requires the matrix to be scaled by $1 + \delta_s$. Hence we must either estimate δ_s, which is not discussed in the original work, or calculate it explicitly, which means we might as well directly solve the ℓ_0 problem.

We also discuss what happens if we vary γ from one iteration to the next. We have shown that doing so can dramatically decrease the error rate in reconstruction. It is ongoing work to see if there is a good way to estimate the optimal γ in terms of minimising the residual, without directly knowing the residual.

Another important difference to the IHT algorithm is that we discuss what happens if we threshold with a value that is strictly larger than the sparsity of the signal we are trying to reconstruct. We often see that choosing a larger thresholding value dramatically improves the success rate at the expense of increasing the number of iterations required for convergence. One option would be to also adaptively vary the thresholding level.

6.8.2 Comparison to Gradient Pursuit

The algorithm we propose, appears on the surface to be very similar to the Gradient Pursuit algorithms discussed in Section 5.1.3 and (Blumensath and Davies 2008). Indeed, the directional update in gradient pursuit is the same as for both IHT and MFR, but the big difference is in how the sparsity constraint is enforced. For the gradient pursuit algorithms, a new dictionary element is added at every iteration, and once added, cannot be removed. In contrast, IHT and MFR make use of a pruning step, so at every iteration we keep only the most important (decided by the largest magnitude) dictionary elements, thus elements can be both added and removed.

6.8.3 Comparison to CoSaMP

Incorporating a least squares step into the MFR algorithm makes it look similar to the CoSaMP algorithm in (Needell and Tropp 2008). There are several significant differences though.

The CoSaMP algorithm solves the least squares problem over the vectors that have support size $3s$ where s is the sparsity of the true solution. The MFR algorithm typically considers a much smaller support size.

Column or support selection for the CoSaMP algorithm is performed by merging the support of the the previous iterate and the "signal proxy", that is $\Phi^{\mathsf{T}}\Phi\mathbf{r}^{(k)} = \Phi^{\mathsf{T}}\Phi\left(\mathbf{x} - \mathbf{x}^{(k)}\right)$. In the case of the MFR algorithm, the frame reconstruction algorithm is used to select the support on which to solve the least squares problem.

As with IHT, the analysis of CoSaMP does not discuss what happens if the sparsity level is unknown or estimated incorrectly. Our results explicitly show that overestimating the sparsity can dramatically increase performance.

Part IV

Discussion

Overview of Algorithms

In this thesis we have discussed many different reconstruction algorithms, in this chapter we will attempt to give a brief overview of some of their properties and how they perform.

7.1 Comparison

The first papers published in the area of compressive sensing showed how ℓ_1 minimisation was a viable alternative to ℓ_0, and under what circumstances they would return the same answer. The algorithms proposed since then offer the following improvements over ℓ_1; (a) higher probability of returning the true solution or (b) faster runtime.

In Table 7.1 we list the algorithms we have mentioned in this thesis. This table omits model based compressive sensing as this is a modification that can be applied to several algorithms, such as CoSaMP, IHT and MFR.

MATCHING PURSUIT		ITERATIVE THRESHOLDING		MISCELLANEOUS	
Matching Pursuit	§5.1	IHT	§5.2.1	ℓ_1 minimisation	§2.4
Orthogonal MP	§5.1.1	Soft Thresholding	§5.2.2	IRLS	§5.4
Stagewise Orthogonal MP	§5.1.2	MFR	§6.2	Belief Propagation	§5.5
Gradient Pursuit	§5.1.3	MFR Adaptive	§6.5		
CoSaMP	§5.1.4	MFR Least Squares	§6.6		

Table 7.1: Categorisation of reconstruction algorithms.

What is important to remember is that not all of these algorithms are directly comparable, many make different assumptions about the measurement matrix. Also the convergence proofs of the reconstruction algorithms often make use of properties of the measurement matrices that are

not easily comparable.

Matching Pursuit, StOMP and GP will work with almost any measurement matrix. CoSaMP, MFR, OMP and ℓ_1 minimisation work with general measurement matrices that satisfy various RIP properties. IHT will also work with a generic matrix, but requires it to be scaled by $1 + \delta_s$ which in practice is difficult to do. A listing of some of these conditions can be seen in Table 7.2.

ALGORITHM	CONDITIONS
$\arg\min_{\hat{\mathbf{x}}:\ \mathbf{y}=\Phi\hat{\mathbf{x}}} \|\hat{\mathbf{x}}\|_0$	Success if $\delta_{2s} < 1$
$\arg\min_{\hat{\mathbf{x}}} \|\mathbf{y} - \Phi\hat{\mathbf{x}}\|_1$	Success if $\delta_{3s} + 3\delta_{4s} < 2$
$\arg\min_{\hat{\mathbf{x}}:\ \mathbf{y}=\Phi\hat{\mathbf{x}}} \|\hat{\mathbf{x}}\|_1$	Success if $\delta_s + \theta_{s,s} + \theta_{s,2s} < 1$
$\arg\min_{\hat{\mathbf{x}}:\ \mathbf{y}=\Phi\hat{\mathbf{x}}} \|\hat{\mathbf{x}}\|_1$	If $m \geqslant C_0 \cdot s \cdot \mu^2(U) \cdot \log\left(\frac{n}{\delta}\right)$ and $m \geqslant C_1 \cdot \log^2\left(\frac{n}{\delta}\right)$ then $\mathbb{P}(\text{success}) \geqslant 1 - \delta$
Matching Pursuit	Recovers $\mathbf{x}$ if $\mathbf{x}$ is a linear combination of rows of Φ
Orthogonal Matching Pursuit (OMP) I	If $\Phi \sim \mathcal{N}\left(0, \frac{1}{m}\right)$ and $m \geqslant C \cdot s \cdot \log\left(\frac{n}{\delta}\right)$ then $\mathbb{P}(\text{success}) \geqslant 1 - 2\delta$
Orthogonal Matching Pursuit (OMP) II	If Φ admissible : $\lambda_{\min}(\Phi^{\mathsf{T}}\Phi) \geqslant \frac{1}{4}$ and $m \geqslant C \cdot s \cdot \log\left(\frac{n}{\delta}\right)$ then $\mathbb{P}(\text{success}) \geqslant 1 - \delta$
Stagewise Orthogonal Matching Pursuit (StOMP)	
Gradient Pursuit (GP)	
CoSaMP	If $\delta_{2s} \leqslant c$ then $\left\|\mathbf{r}^{(k)}\right\|_2 \leqslant 2^{-k} \|\mathbf{x}\|_2 + 20 \|\mathbf{e}\|_2$
Iterative Hard Thresholding (IHT)	Scaled: $\Phi = \frac{\hat{\Phi}}{1+\delta_s}$. If $\beta_{3s} \leqslant \frac{1}{\sqrt{32}} \left(\Rightarrow \delta_{3s} \lesssim 0.1\right)$ then $\left\|\mathbf{r}^{(k)}\right\|_2 \leqslant 2^{-k} \|\mathbf{x}\|_2 + 5 \|\mathbf{e}\|_2$
Soft Thresholding	$\lambda_{\max}(\Phi^{\mathsf{T}}\Phi) < 1$
IRLS	Φ has NSP
Belief Propagation (BP)	$\Phi \sim \{0, \pm 1\}$
Modified Frame Reconstruction (MFR)	(a) $\gamma \geqslant \frac{1}{\delta_{s+2\hat{s}} - \delta_{s+\hat{s}} + 1}$ and $\gamma\delta_{s+2\hat{s}} \leqslant \frac{1}{\sqrt{32}}$ (b) $\gamma < \frac{1}{\delta_{s+2\hat{s}} - \delta_{s+\hat{s}} + 1}$ and $\gamma(1 - \delta_{s+\hat{s}}) \geqslant 1 - \frac{1}{\sqrt{32}}$ or (c) $\frac{3}{4(1-\delta_{s+\hat{s}})} < \gamma < \frac{1}{1-\delta_{s+\hat{s}}}$ and $\delta_{s+\hat{s}} < 1$, then $\left\|\mathbf{r}^{(k)}\right\|_2 \leqslant 2^{-k} \|\mathbf{x}\|_2 + 4\gamma\sqrt{1 + \delta_{s+\hat{s}}} \|\mathbf{e}\|_2$

Table 7.2: Overview of reconstruction algorithms. We consider the model $\mathbf{y} = \Phi\mathbf{x}$, $\Phi \in \mathbb{R}^{m \times n}$ and $\mathbf{x}$ is s-sparse, in particular, we consider the model where there is no noise in the signal.

One point that should be emphasised is that all of these conditions on success are *sufficient* for returning the correct solution, *not* necessary conditions for reconstruction. We believe that in many cases the theorems regarding correct reconstruction severely understate the performance of the algorithms.

For instance, the algorithms we have simulated: CoSaMP and MFR are both able to reconstruct signals which the theoretical results do not suggest that they should be able to. It might be possible to come up with further theorems regarding their performance that give better sufficient

conditions on convergence.

7.2   Simulated Performance

In this section we simulate the ℓ_1 minimisation, CoSaMP and MFR reconstruction algorithms using MATLAB. For all simulations we generate 100 matrices of size $m \times n$ with i.i.d. Gaussian entries of variance $\frac{1}{m}$. Then for a given sparsity level s, for each matrix we generate a vector $\mathbf{x}$ with s non-zero entries where the non-zero entries are generated from the distribution $\mathcal{N}(0, 1)$. Thus each algorithm operates on the same data. Each algorithm then produces, if it can, a vector $\hat{\mathbf{x}}$. If $\hat{\mathbf{x}}$ has the same support as the correct solution $\mathbf{x}$, we say that the algorithm has succeeded. Although not shown here, this has little effect on the results except in the case of the ℓ_1 minimisation algorithm, where the performance is greatly increased.

In Figures 7.1-7.2 we plot the reconstruction for these algorithms and matrices with dimensions given by $n = 400, 800$ and $\frac{m}{n} = \frac{1}{8}, \frac{1}{4}, \frac{3}{8}$ and $\frac{1}{2}$.

Here we see that CoSaMP performs marginally better when $\frac{m}{n} \leqslant \frac{1}{4}$, but with more measurements, the MFR suite of algorithms performs significantly better than CoSaMP. The two variants of MFR, MFR with LS and adaptive step-length consistently outperform the plain version. The least squares variant performs better than the adaptive choice of γ in the cases of more measurements, but when there are fewer measurements, they perform similarly. It should also be noted that the performance of the CoSaMP algorithm drops off quickly, but for MFR, the reconstruction rate slowly drops off with increased sparsity.

As a rule of thumb we offer the following guidelines when using a measurement matrix with Gaussian entries:

- ▶ Use CoSaMP when there is a "small" number of measurements, e.g. when $\frac{m}{n} \lesssim \frac{1}{4}$.
- ▶ Use MFR with LS when $\frac{m}{n} \gtrsim \frac{1}{4}$.
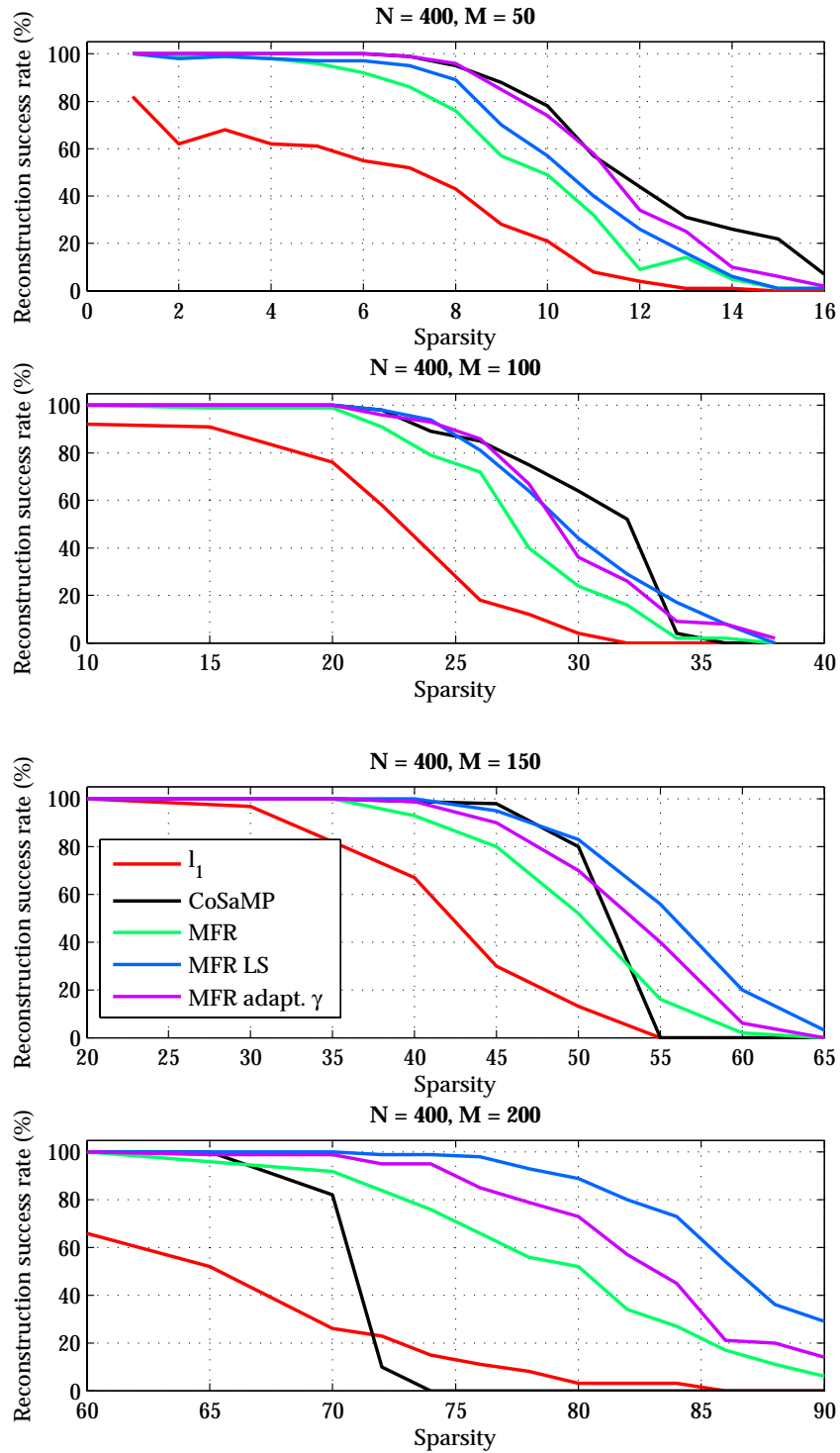
Figure 7.1: Simulated results for several algorithms using matrix sizes with $n = 400$ and $m = 50, 100, 150, 200$.
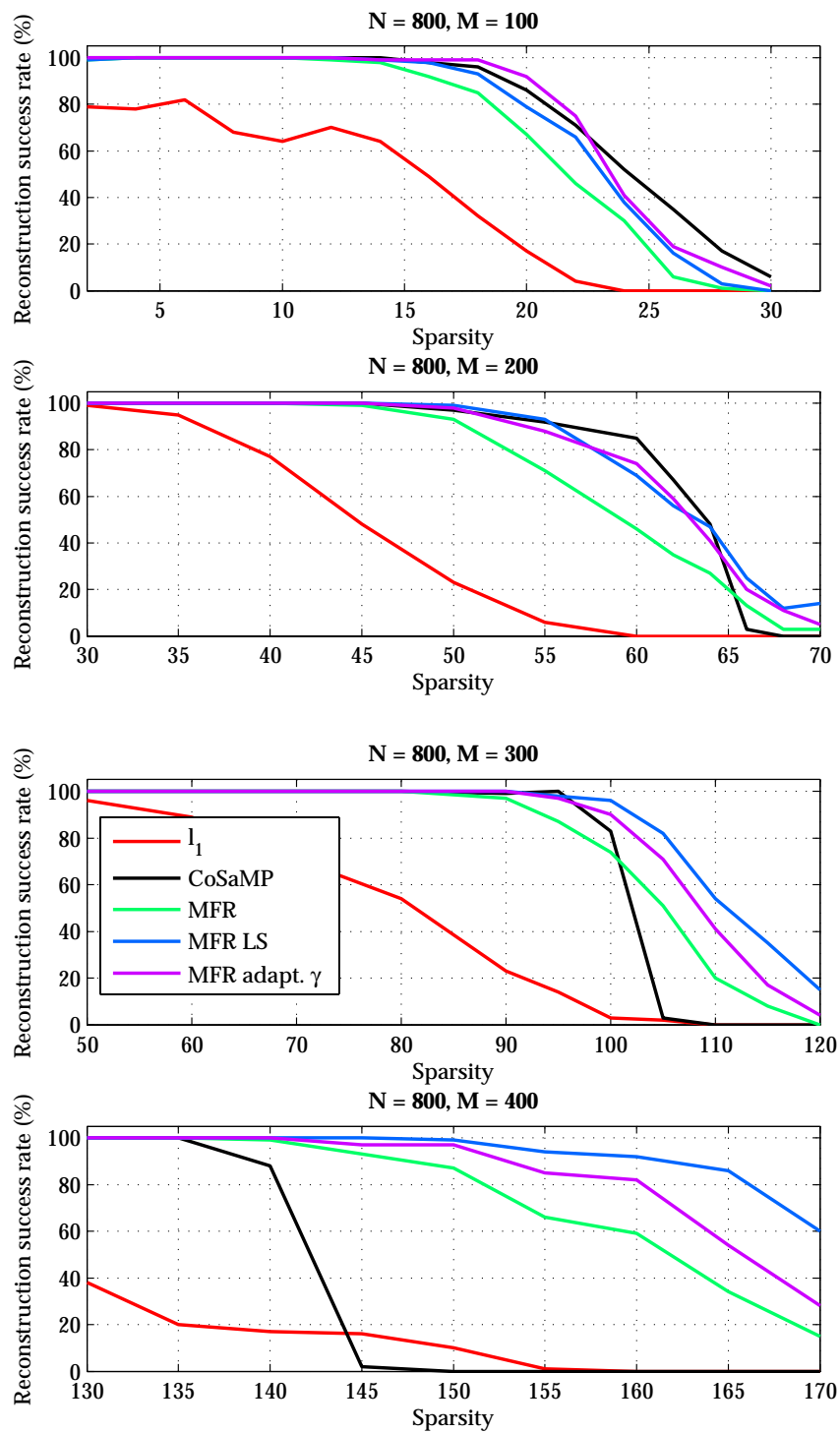
Figure 7.2: Simulated results for several algorithms using matrix sizes with $n = 800$ and $m = 100, 200, 300, 400$.

Restricted Isometry Property and Constants

So far, almost any theorem we have introduced regarding the convergence of a reconstruction algorithm requires the RIP constants δ_s or the modified versions β_s to be bounded quite tightly. The obvious questions are then:

▶ When do these conditions apply?

▶ How can we calculate the restricted isometry properties of a matrix?

We will attempt to shed some light on these questions in this chapter.

8.1 Calculating RIP Constants

Recall the definition of the RIP constants. We say that $\delta_s \geqslant 0$ is the restricted isometry constant of order s is the smallest value such that

$$1 - \delta_s \leqslant \frac{\|\Phi_\Gamma \mathbf{x}\|_2^2}{\|\mathbf{x}\|_2^2} \leqslant 1 + \delta_s. \tag{8.1}$$

holds for all vectors s-sparse vectors $\mathbf{x} \in \mathbb{R}^n$. From Lemma 3.3 we know that the restricted isometry constants are highly related to the eigenvalues of the matrix Φ. Let $\lambda_{\max}(\Phi_\Gamma^\mathsf{T} \Phi_\Gamma)$ and $\lambda_{\min}(\Phi_\Gamma^\mathsf{T} \Phi_\Gamma)$ be the maximum and minimum eigenvalues of the matrix Φ_Γ, then we have

$$1 - \delta_s \leqslant \lambda_{\min}(\Phi_\Gamma^\mathsf{T} \Phi_\Gamma) \leqslant \frac{\|\Phi_\Gamma \mathbf{x}\|_2^2}{\|\mathbf{x}\|_2^2} \leqslant \lambda_{\max}(\Phi_\Gamma^\mathsf{T} \Phi_\Gamma) \leqslant 1 + \delta_s, \tag{8.2}$$

for all index sets Γ and vectors $\mathbf{x}_\Gamma \in \mathbb{R}^t$ for $t \leqslant s$. Indeed the only known way of calculating the RIP constants is by calculating the eigenvalues of the matrices $\Phi_\Gamma^\mathsf{T} \Phi_\Gamma$ for all sets $\Gamma, |\Gamma| = s$, of which there are $\mathcal{O}(n^s)$ such matrices. In fact this is computationally equivalent to directly solving the ℓ_0 minimisation problem $\arg\min_{\hat{\mathbf{x}}:\ \mathbf{y} = \Phi\hat{\mathbf{x}}} \|\hat{\mathbf{x}}\|_0$ which is also of order n^s if the sparsest

solution has sparsity s. Hence if we wish to accurately calculate the RIP constants for a particular matrix to tell if we can apply a theorem, we might as well directly solve the ℓ_0 minimisation problem.

Fortunately we have several theorems telling us when the solution to ℓ_1 and ℓ_0 minimisation problem have the same solution. One of the basic principles of compressive sensing is that if

$$m \geqslant C \cdot s \cdot \log n, \tag{8.3}$$

(where C is a constant depending through the coherence factor $\mu(\Phi, \Psi)$ on the type of measurement basis and the sparsity basis), then the solution to

$$\min_{\hat{\mathbf{x}} \in \mathbb{R}^n} \|\hat{\mathbf{x}}\|_1 \qquad \text{subject to} \qquad \mathbf{y} = \Phi \hat{\mathbf{x}}, \tag{8.4}$$

is exact with overwhelming probability. In particular the probability of success exceeds $1 - \varepsilon$ if

$$m \geqslant C \cdot s \cdot \log\left(\frac{n}{\varepsilon}\right), \tag{8.5}$$

as in (Candès and Romberg 2007). Work in (Donoho and Tanner 2006), quoted in (Needell and Tropp 2008) says that we can recover most sparse signals via ℓ_1 minimisation if $m \approx 2s \log n$, or equivalently $s \approx m/(2 \log n)$. See Table 8.1 for an illustration of typical m, n and maximum s values according to this formula.

				n			
m	100	200	400	800	1600	3200	6400
50	5.4	4.7	4.2	3.7	3.4	3.1	2.9
100		9.4	8.3	7.5	6.8	6.2	5.7
200			16.7	15.0	13.6	12.4	11.4
400				29.9	27.1	24.8	22.8
800					54.2	49.6	45.6
1600						99.1	91.3

Table 8.1: Largest sparsity value for which we can reconstruct the original signal with overwhelming probability using ℓ_1 minimisation.

The problem with these results though is that it does not give us the δ_s values which we require for algorithms such as IHT (Theorem 5.11). Results such as Lemma 3.1 lower the bound the probability that the RIP constants are bounded above by a particular value, which means that we know, in general, when an algorithm is applicable. The problem however is that this bound is not sufficiently tight and the lower bound for the probability can be too low to be meaningful (indeed, it can degenerate to a statement of the form "the probability is at least $-10^{100} \ll 0$"!).

It is however possible to lower bound the RIP constants using Monte Carlo methods. If we randomly select columns from the matrix in question and then calculate the singular values, we know that the RIP constant must be at least as large. A complete description of the algorithm can be seen in Algorithm 8.1. Figure 8.1 was produced using this method.

Algorithm 8.1 Monte Carlo method for RIP Constants

Input:

▶ A matrix $\Phi \in \mathbb{R}^{m \times n}$.
▶ A sparsity level s.

Output:

▶ A lower bound on the RIP constant δ_s of order s.

1: $\delta_s \leftarrow 0$
2: **for** $k = 0, \dots, \infty$ **do**
3:     Select a set of columns Γ of Φ uniformly at random
4:     $\delta_s \leftarrow \max \left\{ \delta_s, \, \lambda_{\max} \left(\Phi_\Gamma^\mathsf{T} \Phi_\Gamma \right) - 1, \, 1 - \lambda_{\min} \left(\Phi_\Gamma^\mathsf{T} \Phi_\Gamma \right) \right\}$
5: **end for**
6: **return** δ_s

8.2   When do the theorems apply?

So what values of m, n and s are suitable for applying the convergence theorems? The figures in Table 8.1 paint a very dismal view of when we can recover a signal via ℓ_1 minimisation and if we compare these to our results in Figures 7.1-7.2, we see that in fact we are simulating with significantly higher sparsity values (i.e. less sparse signals) than in the table. Yet we are able to effect reconstruction.

Both the IHT and the CoSaMP algorithm have very similar conditions on the RIP constants for convergence. Theorem 5.4 about CoSaMP makes use of the fact that $\delta_{4s} \leqslant 0.1$ and Theorem 5.11 says that if

$$\beta_{3s} \leqslant \frac{1}{\sqrt{32}} \qquad \Longleftrightarrow \qquad \delta_{3s} \leqslant \frac{\sqrt{2}}{16 - \sqrt{2}} \approx 0.097, \tag{8.6}$$

then the IHT algorithm will converge to the correct s-sparse solution. Unfortunately it is difficult to compare these results to our theorem about MFR as we have a difference of two $\delta_{(\cdot)}$ terms, something which is difficult to conceptualise.

Using our Monte Carlo method for lower bounding the RIP constants (Algorithm 8.1) we created Figure 8.1 showing the spread of δ_s values for 100 random matrices of size 200 by 400 for various sparsity levels. In fact we see that even for $s = 10, m = 200$ and $n = 400$ we get $\delta_s \geqslant 0.6$, much larger than the $\delta_{3s} \leqslant 0.1$ required for IHT or even $\delta_{4s} \leqslant 0.5$ which is required for ℓ_1 minimisation, to reconstruct any s-sparse signal.

So the cases we simulate have a sufficiently large s value so that they are *not* covered by the theorems regarding convergence of the reconstruction algorithms. But what is interesting, is that here we can see that our algorithm, as well as CoSaMP performs significantly better than ℓ_1 minimisation. If we only look at the regions where ℓ_1 is guaranteed to work (albeit only with high probability) we learn nothing interesting, as all the algorithms work.
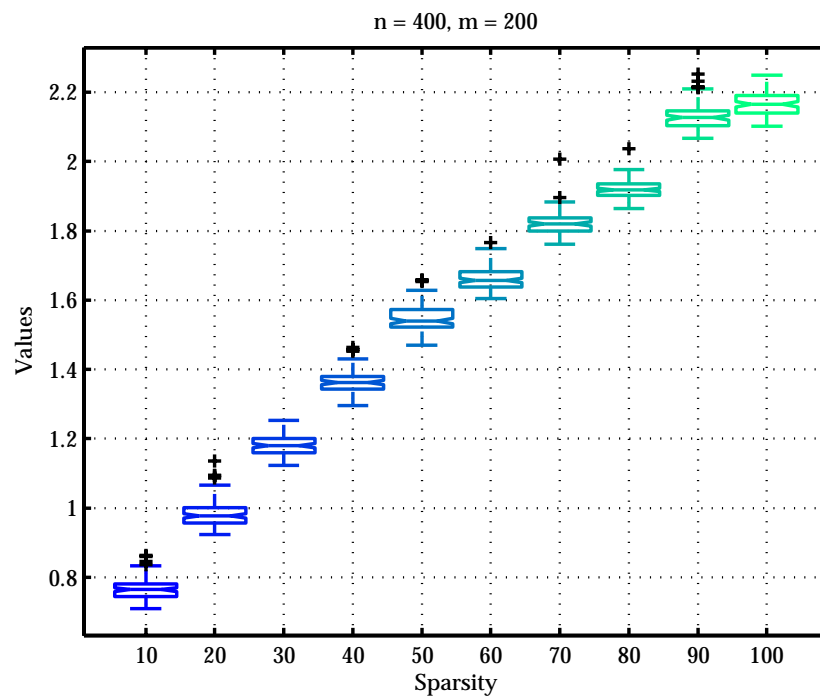
Figure 8.1: "Box-and-whiskers" plot of lower bounds for δ_s values for 200×400 matrices. Results calculated for 100 different Gaussian matrices with i.i.d. entries $\mathcal{N}\left(0, \frac{1}{m}\right)$ by randomly selecting s columns 10,000 times.

9

Conclusion

In this thesis we have given an overview of compressive sensing and frames. We have discussed the fundamentals of compressive sensing, from designing measurement matrices to conditions guaranteeing successful reconstruction via ℓ_1 minimisation.

We have then explored a number of alternatives to ℓ_1 minimisation including matching pursuit algorithms and iterative algorithms, as well as some of the more arcane methods such as belief propagation.

On top of this we have presented a new algorithm, MFR, for which we prove some convergence properties, as well as giving two variants on this algorithm both of which offer significant improvements in reconstruction and convergence. Our simulations show that the MFR algorithm outperforms other algorithms when utilising a relatively high number of measurements.

Finally we present a brief overview of all the algorithms mentioned, as well as a discussion on when the various convergence and reconstruction theorems can be applied.

Index

Bibliography

Abramovich, F., Y. Benjamini, D. L. Donoho, and I. M. Johnstone (2000). Adapting to unknown sparsity by controlling the false discovery rate. *Annals of Statistics*.

Alliney, S. and S. Ruzinsky (1994, March). An algorithm for the minimization of mixed ℓ_1 and ℓ_2 norms with application to Bayesian estimation. *IEEE Trans. Signal Processing 42*, 618–627.

Baraniuk, R., M. Davenport, R. DeVore, and M. Wakin (2007). A simple proof of the restricted isometry property for random matrices. *Constructive Approximation*.

Baraniuk, R. G., V. Cevher, M. F. Duarte, and C. Hegde (2008). Model-based compressive sensing. *CoRR abs/0808.3572*.

Baron, D., M. Duarte, S. Sarvotham, M. B. Wakin, and R. G. Baraniuk (2006). Distributed compressed sensing. *Preprint*.

Blumensath, T. and M. Davies (2008, June). Gradient pursuits. *Signal Processing, IEEE Transactions on 56*(6), 2370–2382.

Blumensath, T. and M. E. Davies (2007). Sampling theorems for signals from the union of linear subspaces. *Preprint*.

Blumensath, T. and M. E. Davies (2008, May). Iterative Hard Thresholding for Compressed Sensing. *ArXiv e-prints 805*.

Blumensath, T. and M. E. Davies (2008a). Stagewise weak gradient pursuits part i: Fundamentals and numerical studies. *Preprint*.

Blumensath, T. and M. E. Davies (2008b). Stagewise weak gradient pursuits part ii: Theoretical properties. *Preprint*.

Boyd, S. and L. Vandenberghe (2004, March). *Convex Optimization*. Cambridge University Press.

Candès, E. and J. Romberg (2007). Sparsity and incoherence in compressive sampling. *Inverse Problems 23*(3), 969–985.

Candès, E. J., J. Romberg, and T. Tao (2006a, Feb.). Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *Information Theory, IEEE Transactions on 52*(2), 489–509.

Candès, E. J., J. Romberg, and T. Tao (2006b). Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics 59*(8), 1207–1223.

Candès, E. J., M. Rudelson, T. Tao, and R. Vershynin (2005). Error correction via linear programming. In *FOCS*, pp. 295–308.

Candès, E. J. and T. Tao (2005, Dec.). Decoding by linear programming. *Information Theory, IEEE Transactions on 51*(12), 4203–4215.

Candès, E. J. and T. Tao (2006, Dec.). Near-optimal signal recovery from random projections: Universal encoding strategies? *Information Theory, IEEE Transactions on 52*(12), 5406–5425.

Cohen, A., W. Dahmen, and R. DeVore (2008). Compressed sensing and best k-term approximation. *J. American Mathematical Society*.

Crouse, M. S., R. D. Nowak, and R. G. Baraniuk (1998, April). Wavelet-based statistical signal processing using hidden markov models. *IEEE Transactions on Signal Processing 46*, 886–902.

Daubechies, I., M. Defrise, and C. D. Mol (2004). An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure and Applied Mathematics 57*(11), 1413–1457. 10.1002/cpa.20042.

Daubechies, I., R. DeVore, M. Fornasier, and C. S. Gunturk (2008). Iteratively re-weighted least squares minimization for sparse recovery.

Daubechies, I., M. Fornasier, and I. Loris (2007). Accelerated projected gradient method for linear inverse problems with sparsity constraint. *The Journal of Fourier Analysis and Applications*.

Davis, G., S. G. Mallat, and M. Avellaneda (1997). Greedy adaptive approximation.

DeVore, R. and V. N. Temlyakov (1996). Some remarks on greedy algorithms.

Donoho, D., I. Drori, V. Stodden, Y. Tsaig, and M. Shahram (2007). SparseLab. Technical report, `http://sparselab.stanford.edu/`.

Donoho, D. L. (2006, April). Compressed sensing. *IEEE Transactions on Information Theory 52*(4), 1289–1306.

Donoho, D. L. and J. Tanner (2006, July). Counting faces of randomly projected polytopes when the projection radically lowers dimensions. Technical Report 2006-11, Department of Statistics, Stanford University.

Donoho, D. L., Y. Tsaig, I. Drori, and J.-L. Starck (2007). Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit. *Submitted IEEE Transactions on Information Theory*.

Efron, B. (1979). Bootstrap methods: Another look at the jackknife. *The Annals of Statistics 1*, 1–26.

Efron, B. (1981). Nonparametric estimates of standard error: The jackknife, the bootstrand and other methods. *Biometrika 68*, 589–599.

Eldar, Y. and M. Mishali (2008). Robust recover of signals from a union of subspaces. *Preprint*.

Figueiredo, M., R. Nowak, and S. Wright (2007, Dec.). Gradient projection for sparse recon-struction: Application to compressed sensing and other inverse problems. *Selected Topics in Signal Processing, IEEE Journal of 1*(4), 586–597.

Friedman, J. H. and W. Stuetzle (1981). Projection pursuit regression. *Journal of the American Statistical Association 76*(376), 817–823.

Gallager, R. G. (1963). *Low Density Parity Check-Codes*. Ph. D. thesis, MIT, Cambridge, MA, USA.

Gilbert, A. C., M. J. Strauss, J. A. Tropp, and R. Vershynin (2007). One sketch for all: fast al-gorithms for compressed sensing. In *STOC '07: Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, New York, NY, USA, pp. 237–246. ACM.

Golub, G. H. and F. V. Loan (1996). *Matrix Computations* (3rd ed.). John Hopkins University.

Gröchenig, K. (1993, Dec). Acceleration of the frame algorithm. *Signal Processing, IEEE Trans-actions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on] 41*(12), 3331–3340.

Heil, C. E. and D. F. Walnut (1989). Continuous and discrete wavelet transforms. *SIAM Re-view 31*(4), 628–666.

Johnson, W. B. and J. Lindenstrauss (1984). Extensions of lipschitz mappings into a hilbert space. *Proc. Conf. Modern Anal. Prob 26*, 189–206.

Lawson, C. L. (1961). *Contributions to the Theory of Linear Least Maximum Approximation*. Ph. D. thesis, University of California, Los Angeles.

Lorentz, G. G., M. Golitschek, and Y. Makovoz (1996). *Constructive Approximation: Advanced Problems*, Volume 304. Springer-Berlin.

Lu, Y. M. and M. N. Do (2008, March). Sampling signals from a union of subspaces. *IEEE Signal Processing Magazine 25*, 41–47.

Mallat, S. G. (1999). *A Wavelet Tour of Signal Processing*. San Diego: Academic Press.

Mallat, S. G., G. Davis, and Z. Zhang (1994, July). Adaptive time-frequency decompositions. *SPIE Journal of Optical Engineering 33*, 2183–2191.

Mallat, S. G. and Z. Zhang (1993, Dec). Matching pursuits with time-frequency dictionaries. *Signal Processing, IEEE Transactions on 41*(12), 3397–3415.

Mendelson, S., A. Pajor, and N. Tomczak-Jaegermann (2006). Uniform uncertainty principle for bernoulli and subgaussian ensembles.

Needell, D. and J. A. Tropp (2008). Cosamp: Iterative signal recovery from incomplete and inaccurate samples.

Nyquist, H. (1928, February). Certain topics in telegraph transmission theory. *Transactions of the A.I.E.E.*, 617–644.

Osborne, M. R. (1985). *Finite Algorithms in Optimization and Data Analysis*. Chichester: John Wiley & Sons Ltd.

Pati, Y. C., R. Rezaiifar, and P. S. Krishnaprasad (1993). Orthogonal matching pursuit: Recur-sive function approximation with applications to wavelet decomposition.

Rockafellar, R. T. (1970). *Convex Analysis*. Princeton, NJ: Princeton University Press.

Sarvotham, S., D. Baron, and R. G. Baraniuk (2006, July). Compressed sensing reconstruction via belief propagation. Technical Report Technical Report ECE-0601, Department of Electrical and Computer Engineering, Rice University, Houston, USA.

Shannon, C. E. (1949, January). Communication in the presence of noise. *Proc. Institute of Radio Engineers 37*(1), 10–21.

Stojnic, M., F. Parvaresh, and B. Hassibi (2008). On the reconstruction of block-sparse signals with an optimal number of measurements. *Preprint*.

Tropp, J. A. (2004, Oct.). Greed is good: algorithmic results for sparse approximation. *Information Theory, IEEE Transactions on 50*(10), 2231–2242.

Tropp, J. A. and A. C. Gilbert (2007, Dec.). Signal recovery from random measurements via orthogonal matching pursuit. *Information Theory, IEEE Transactions on 53*(12), 4655–4666.

Verdú, S. (1998). *Multiuser Detection*. Cambridge University Press.

Wakin, M. B., S. Sarvotham, M. F. Duarte, D. Baron, and R. G. Baraniuk (2005, April). Recovery of jointly sparse signals from few random projections. *Proc. Workshop on Neural Info. Proc. Sys. 86*, 572–588.

Zermelo, E. (1904). Beweis, dass jede menge wohlgeordnet weden kann. *Mathematische Annalen 59*, 514–16.