# LOST IN SPARSITY:
# COMPRESSED SENSING
# AND ITS APPLICATIONS

WILLIAM HARTOG

ADVISORS:
SUBHABRATA SEN
CLIFFORD TAUBES

### Acknowledgements

First I would like to thank my primary advisor Subhabrata Sen for being a consistently great source of knowledge and support throughout the past year, and my co-advisor Clifford Taubes for fitting reading through my thesis into his busy schedule.

I would also like to thank Patrick Lopatto, who initially recommended I look into compressed sensing for my thesis topic, and Joe Blitzstein for introducing me to statistics through three fantastically-taught courses.

Finally, I would like to thank my friends and family, especially my parents, for their love and support.

ABSTRACT. The aim of this paper is to overview the history and theory of compressed sensing, a relatively new but powerful field that allows for practical and fast signal processing for signals which are sparse in some basis. We provide a geometric intuition for this requirement, as well as theoretical guarantees under which sparse recovery via $\ell^1$-minimization succeeds. We also give algorithms for the implementation of the method, a set of simulations comparing the recovery success of various algorithmic settings, and an example of applying compressed sensing to the application area of medical imaging.

## CONTENTS

INTRODUCTION

A common mathematical and statistical problem, which is introduced in high school coursework and yet continually provides new challenges for experienced researchers, is that of solving a linear system of equations. A general setting we may consider involves a measurement matrix $\mathbf{A} \in \mathbb{C}^{m \times N}$, a parameter vector or signal $\mathbf{x} \in \mathbb{C}^N$, and a response vector $\mathbf{y} \in \mathbb{C}^m$, satisfying

$$(0.1) \qquad \mathbf{Ax} = \mathbf{y}.$$

For example, in the world of signal processing for image compression, which turns out to be a topic that we will very much care about, the signal $\mathbf{x}$ would be a vector comprising the parameters of the original image that we want to recover, $\mathbf{y}$ is the compressed version of this image, and $\mathbf{A}$ is a linear transformation mapping the image vector $\mathbf{x}$ to the compressed response vector $\mathbf{y}$ through the relationship (0.1).

As such the principal problem arises when $\mathbf{x}$ is unknown but $\mathbf{A}$ and $\mathbf{y}$ are known, which boils down to an inverse problem, as is the case for the aforementioned image compression example. Such problems form the basis (pun intended) for row reduction in introductory linear algebra coursework, as well as for statistical inference in the setting of least-squares regression, where the signal $\mathbf{x}$ is interpreted as the estimated effects of a set of parameters and $\mathbf{y}$ is taken to have some error term which is usually Gaussian (alternatively known as Normal) [1]. Moreover, this setting is exceedingly important from both mathematical and statistical perspectives because of the prevalence of this model in application, meaning solving difficult variations of the problem can often produce technologically useful results.

For methods such as these, success, as defined by the recovery of a unique exact solution or estimate for $\mathbf{x}$, can be expressed in terms of the relationship between the number of parameters, $N$, and the number of observations, $m$. Linear regression, in particular, is only a well-posed problem when the number of observations is greater than the number of solutions, and the linear system (0.1) could possibly have a unique solution for $m \geq N$, provided certain conditions. However, the solution space $\{\mathbf{z} : \mathbf{Az} = \mathbf{y}\}$ for the linear system (0.1) will be infinite if $m < N$, meaning exact recovery of a particular $\mathbf{x}$ requires additional specifications.

While this difficulty might appear a deterrent, a natural motivation to further investigate the $m < N$ case is that in application the ability to successfully recover signals with fewer measurements is much more practical, especially as the dimension of the parameter vector increases. Such time- and resource-saving measures have always been significant, but especially in the modern age of big data and large-scale computation, minute reductions in the number of measurements scale quickly and are instrumental to the production of cost-efficient technology. As such, the pursuit of theoretical minimums on the number of observations for such a system has been an ongoing object of scientific inquiry.

A common intuitive explanation for the ability to obtain unique solutions for systems of linear equations goes along the lines of the following adage: "As many equations as unknowns". In our context, when our parameter vector $\mathbf{x}$ has fixed length $N$, and we want to decrease the number of measurements $m$, there is no way to achieve this balance if we treat every parameter in $\mathbf{x}$ as a separate unknown. The crucial realization here is that most data follows a certain structure. For example if every row vector in the set of row vectors $\{\mathbf{A}_i\}_{i \in [m]}$ can be expressed in some set of linearly independent vectors $\{\mathbf{b}_j\}_{j \in [d]}$ that span a subspace $U \subset \mathbb{C}^N$ of dimension $d < N$.
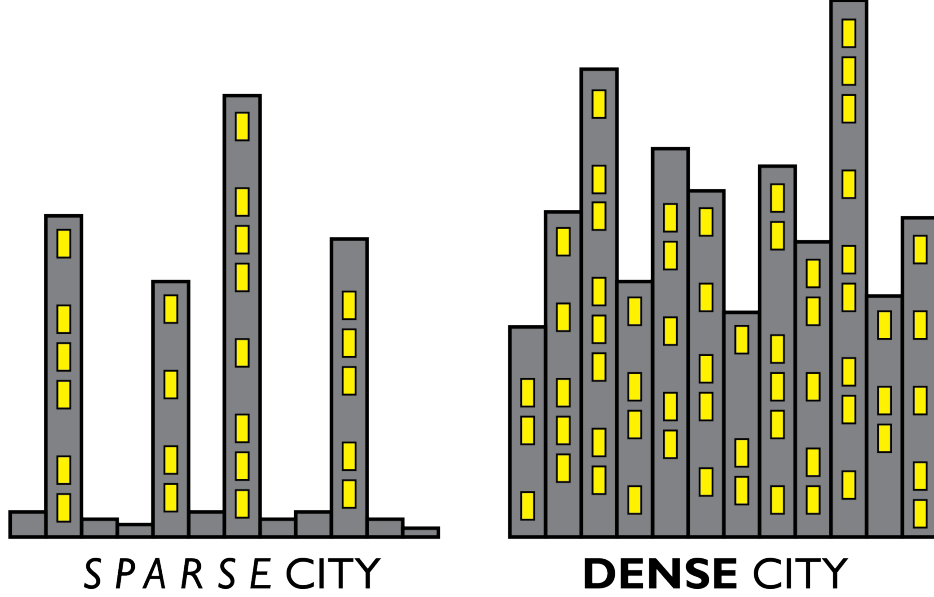
FIGURE 1. Illustration depicting the hypothetical Sparse City and Dense City, whose differences are characterized by the number of buildings with non-negligible heights.

Note that $[d]$ is shorthand for the set of integers $[d] := \{1, 2, ..., d\}$. After adding vectors to form a basis of $\mathbb{C}^N$, we can re-express the measurement matrix with this new basis, and our new parameter vector will only have nonzero entries for the first $d$ components (corresponding with our set of $d$ vectors that span $U$).

This intuitive approach nicely demonstrates how the goal of decreasing the number of measurements $m$ corresponds with the low-dimensional structure of the parameter vector, and thus sparsity of $\mathbf{x}$. Defining $\text{supp}(\mathbf{x}) := \{i \in [N] : \mathbf{x}_i \neq 0\}$ as the support set of indices $i$ for which the $i$th entry of $\mathbf{x}$ is nonzero, we call $s := \text{card}(\text{supp}(\mathbf{x}))$ the *sparsity* of $\mathbf{x}$ (the cardinality or size of the support of $\mathbf{x}$), and we call $\mathbf{x}$ $s$-sparse if $\text{card}(\text{supp}(\mathbf{x})) \leq s$. Of course, it is important to note that in practice signals will not be exactly sparse, and rather have most entries very close to zero, but exact sparsity plays an important role in both major theoretical results and in a high-level understanding of our methods.

Hopefully the previous discussion is helpful to understand the importance of sparsity, but to further visualize why sparsity allows for significantly faster recovery, consider the dichotomy between two cities, represented in terms of the projection of their skylines onto a one-dimensional space. The first city, on the left of Figure 1, is called Sparse City, and is mostly comprised of low rise buildings and a few very tall skyscrapers. A reasonable real world comparison to Sparse City is Los Angeles, which has much fewer skyscrapers than other, shorter buildings. On the other end of the spectrum is Dense City, on the right of Figure 1, in which practically every building is of a sizable height, including many larger skyscrapers. Dense City can be compared to New York City, which is known for its very dense skyline of large buildings.

If asked to exactly reconstruct the heights and locations of all "sizable" buildings in Sparse City and Dense City, it clearly seems doing so for Sparse City would be the easier task, in the sense that less information would be required. To draw a

comparison to our linear system setting, we need a notion of what a "measurement" is, but any way you might take a measurement, including taking various snapshots of the city, the sparse nature and clear structure of Sparse City make it the easier of the two cities to classify.

**The History of Signal Processing and Sampling Theorems.** Armed with a specific target of sparsity through low-dimensional structures in our experiments, we turn to what has historically been a rich pursuit of low-dimensional structures in statistical inquiry.

One reason low-dimensional structures have been a common point of academic interest is a hypothesis that human natural vision is an example of sparse recovery [19]. Natural vision systems are able to display highly accurate representations of their perceived reality incredibly well, in terms of time and efficiency. In particular, potential measurements contain a high amount of extraneous information, and thus a primary purpose of natural vision is to recreate these images while minimizing redundancy in observation.

This natural example of exploitation of low-dimensional structures motivates the investigation by engineers into image processing using computers through a similar exploitation [19]. Of course images hardly represent of the spectrum of real-world data that are sparse in some way, as we see in the history of low-dimensional structures in statistics.

In particular, much of this history is concerned with signals $x(t)$ that are functions of time $t$ and often represent data such as audio signals. Then we will look at the set of signals that are "band-limited", which are formally defined as functions such that their Fourier transform

$$\hat{x}(\omega) = \int_{-\infty}^{\infty} x(t) \exp(-i\omega t) dt$$

vanishes for $\omega$ outside of some interval $[-\Omega, \Omega]$, which translates to the condition that a signal has a maximum frequency of $f_{\max} = \Omega/2\pi$. This may appear an improper structure to investigate if we cared about a discrete signal such as an image with finitely many pixels. However the results on recovery of such signals via sampling are rich when taking finitely many measurements of a signal, for example by taking measurements $x(\frac{n\pi}{2})$ for integers $n$, where $\{\frac{n\pi}{2}\}_n$ are the associated time values. We call $\mathcal{B}_1(\Omega)$ the set of band-limited functions with associated constant $\Omega$.

A characterization of the minimum number of samples required for band-limited signal recovery given in the 1920s is attributed to Harry Nyquist and Claude Shannon (though earlier proposed by E.T. Whittaker in 1915 and later by Vladimir Kotelnikov) [19]:

**Theorem 0.1.** *(Nyquist-Shannon Sampling) To recover a band-limited signal $x(t)$ with $x \in \mathcal{B}_1(\Omega)$, it is necessary to sample the signal at a rate twice the maximum frequency $f_{\max} = \Omega/2\pi$.*

We omit the proof of this theorem, but note that while powerful, it can become rather inefficient, especially when the maximum frequency becomes rather large while the range of frequencies stays the same. Intuitively we would like to obtain the same sampling rate for a given frequency range, regardless of its location in the spectrum [19]. Sampling at what is usually known as the Nyquist sampling rate is costly, and can be potentially dangerous for patients in medical imaging, so an alternative approach would be welcome. This is not at all to understate the value and impact of

the Nyquist sampling scheme, which was so commonly used in signal processing that a major improvement on the Nyquist rate was sure to be a landmark result.

**The Failure of $\ell^0$-Minimization and the Embrace of $\ell^1$-Minimization.** A natural way to approach the problem of sparse recovery is as a minimization problem with respect to the $\ell^0$ norm. The $\ell^0$ norm of $\mathbf{x}$, $\|\mathbf{x}\|_0 = |\operatorname{supp}(\mathbf{x})|$, which is the number of nonzero entries in the vector $\mathbf{x}$, is just the sparsity of $\mathbf{x}$ as we defined earlier. This gives rise to the optimization problem known as $\ell^0$-minimization.

$$(0.2) \qquad \text{minimize } \|\mathbf{z}\|_0 \quad \text{subject to } \mathbf{A}\mathbf{z} = \mathbf{y}.$$

However, $\ell^0$-minimization is known to be an NP-hard problem in the worst case [8], which doesn't bode well for its utility. Specifically, the problem, when generalized to the condition that $\|\mathbf{A}\mathbf{z} - \mathbf{y}\|_2 \leq \eta$ (where $\|\cdot\|_2$ is the $\ell^2$ norm defined as $\|\mathbf{z}\|_2 = \sqrt{\sum_{i=1}^m |\mathbf{z}|_i^2}$ for $\mathbf{z} \in \mathbb{R}^m$) for some fixed $\eta \geq 0$, is NP-hard, which is proven in [8] via reduction to the exact cover problem with 3-sets, a brief description of which is as follows:

**Definition 0.2.** The **Exact Cover by 3-Sets Problem** is to find, for some collection of sets $C = \{\mathcal{C}_i\}_{i \in [N]}$ with $\mathcal{C}_i \subset [m]$ and $|\mathcal{C}_i| = 3$ for all $i \in [N]$, a partition of $[m]$ using elements of $C$, i.e. some index set $J \subset [N]$ with $\{\mathcal{C}_j\}_{j \in J}$ disjoint and $\cup_{j \in J} \mathcal{C}_j = [m]$.

We omit the proof of this reduction, which can be found in [8].

Of course, the notion of NP-hardness is a worst case statement, and the worst case might turn out to be very rare. Nevertheless, knowledge of this worst case scenario and the persisting difficulty of exact computation for the $\ell^0$ norm [9] make a case for looking for alternatives.

We might instead minimize the $\ell^1$, defined as $\|\mathbf{x}\|_1 = \sum_{i=1}^N |\mathbf{x}|_i$, which would produce an output that has a small total sum of absolute entries, which is not as suitable as the $\ell^0$ norm but will turn out to be a sufficient proxy. Thus, while $\ell^0$-minimization is a natural consideration, it is not feasible as a method, either in theory or in practice, and as such we will focus our attention on the looser but more implementable $\ell^1$-minimization, which as we will see has enjoyed historical successes [8, 19].

Apart from the Nyquist rate, band-limited signals played a large role in signal recovery results in the PhD work of Benjamin Logan in the 1960s, which produced the following result regarding $\ell^1$-minimization [11]:

> Suppose we observe a signal $y$ which consists of a band-limited signal $x_0$, superimposed with an error $e_0$ which is sparse in the time domain. If the product of the bandwith of $x_0$ and the size of the support of $e_0$ is less than $\pi/2$, the true band-limited signal can be recovered by $\ell^1$ minimization, no matter how large the error is in magnitude, or where its support is located.

This result, often known as Logan's Phenomenon, characterizes the ability of the method known as $\ell^1$-minimization to recover signals, where $\ell^1$-minimization in the context of Logan's work is the optimization problem

$$\text{minimize } \|x - y\|_1 \quad \text{subject to } x \in \mathcal{B}_1(\Omega),$$

where the observed signals $y$ are obtained from the original signal $x_0$ by some offset error $y = x_0 + e_0$, where $e_0$ is "sparse in the time domain" in the sense that its support

is a set of small measure (which is a rigorous generalization of the geometric concept of area). Here $e_0$ is interpreted as a function $e_0 : \mathbb{R} \to \mathbb{R}$ with input of time $t$. We refer the reader to [11] for a more complete description of the result.

Since the signals $x$ are continuous functions, we interpret $\| \cdot \|_1$ the $\ell^1$ norm differently than on $\mathbb{C}^n$, but Logan's Phenomenon marks a first example of the usefulness of $\ell^1$-minimization as an approach for successfully recovering sparse signals, with more examples to follow in the latter decades of the 20th century [19]. With regard to our main environment of interest, when we want to recover $\mathbf{x} \in \mathbb{C}^N$ given the measurement $\mathbf{y} = \mathbf{Ax}$ where $\mathbf{y} \in \mathbb{C}^m$ and $\mathbf{A} \in \mathbb{C}^{m \times N}$, lasso regression is a particularly popular algorithm, which is formulated as follows (with $\| \cdot \|_2$ the $\ell^2$ norm as defined earlier in the section):

$$\text{minimize}_{\mathbf{z}} \; \|\mathbf{y} - \mathbf{Az}\|_2^2 \quad \text{subject to } \|\mathbf{z}\|_1 \leq k.$$

In fact, lasso is equivalent in formulation to basis pursuit, which is another optimization based on the $\ell^1$ norm [19]:

(0.3) $$\text{minimize } \|\mathbf{z}\|_1 \quad \text{subject to } \mathbf{Az} = \mathbf{y}.$$

Given this clear objective, which is to achieve a more efficient sampling rate than the standard Nyquist rate, and a useful tool, $\ell^1$-minimization (intuition for which we will further develop in Section 1), the ground is set for a major breakthrough.

**Compressed Sensing: The Seminal Work.** With this backdrop of $\ell^1$ norm methods for signal recovery from the latter half of the 20th century, the ball was bound to drop, in the eye of what John Wright and Yi Ma describe as a "perfect storm" [19]. Between 2005 and 2006, through a collaboration betweenEmmanuel Candès, Justin Romberg, Terence Tao, and David Donoho [2, 6], the observation that $\ell^1$-minimization performed recovery of sparse signals with few measurements was formalized into what became known as the field of compressed sensing (CS), and can alternatively be known by any combination of {compressed, compressive} and {sensing, sampling} [8, 19]. Their key results were the ability to recover a sparse signal $\mathbf{x} \in \mathbb{C}^N$ supported on a set of size proportional to $N$, with a number of measurements $m$ that significantly bests the Nyquist rate, through a random measurement matrix $\mathbf{A} \in \mathbb{C}^{m \times N}$.

Precise statements of such results are given in Section 2. In addition, the usage of a random measurement matrix is a crucial development, which we will likewise expound on in later sections. The importance of the work of these academics, and the further development of general results in Candès and Tao's 2006 paper "*Decoding by Linear Programming*" contributed to a veritable explosion of work in the budding field of compressed sensing. By 2010 additional theoretical and applied results were produced on the subject, including multiple applications of the method to industry, including most notably medical resonance imaging (MRI) and the single-pixel camera, but also radar technology, sparse approximation, machine learning, and facial recognition [8, 19]. Since this initial boom, compressed sensing has proven to be a valuable numerical method, especially in the age of big data.

As such, this paper covers three main topics as relating to the field of compressed sensing. First, we provide an overview of the theory behind compressed sensing, namely the method of $\ell^1$-minimization, the conditions under which $\ell^1$-minimization succeeds at sparse recovery, and the need for random matrices (Sections 1 and 2). Second, we highlight a few of the more commonly used algorithms for implementing compressed sensing, including a comparison of their performances on simulated

datasets (Section 3). Third, we give an example of what an application of compressed sensing looks like in practice, in the field of MRI, since the power of the field lies in its applicability, and most of the current challenges in compressed sensing involve its application to real data (Section 4).

## 1. Geometric Intuition for $\ell^1$-Minimization

As mentioned in the Introduction, compressed sensing relies on an optimization method that had been in use for years before the seminal work of Donoho, Candès, Romberg, and Tao, $\ell^1$-minimization (0.3). While the historical success of this method may assuage potential concerns, it remains that the accuracy of $\ell^1$-minimization in sparse recovery is not as clear as that of $\ell^0$-minimization (0.2). Specifically, minimizing the $\ell^1$ norm $\|\mathbf{x}\|_1$ at face value does not produce sparse vectors as outputs, while the whole point of compressed sensing is to like to recover vectors which are sparse.

For this reason we take a brief aside into the geometry of the $\ell^1$ norm and why the problem as constructed is well-formulated.

With regard to the minimization problem (0.3), we know the solution space is

$$\{\mathbf{x}\} + \ker(\mathbf{A}) = \{\mathbf{x} + \mathbf{u} : \mathbf{u} \in \ker(\mathbf{A})\}$$

where the kernel is defined as the set $\ker(\mathbf{A}) = \{\mathbf{u} \in \mathbb{C}^N : \mathbf{A}\mathbf{u} = 0\}$. In all of these cases $\ker(\mathbf{A})$ is a hyperplane in $\mathbb{C}^N$, where we wish to find the point $\mathbf{z} \in S$ on that hyperplane which minimizes its $\ell^1$ norm $\|\mathbf{z}\|_1$. Using a graphical representation adapted from Wright and Ma [19] we can represent this problem in the case of domain $\mathbb{R}^3$ for simplicity, which we show in Figure 2. Here, we can think of starting with the unit $\ell^1$ ball (or some scaled down version such that it does not intersect the feasible set $\{\mathbf{x}\} + \ker(\mathbf{A})$), which appears as a diamond in $\mathbb{R}^3$, and subsequently increasing the radius of the ball until it first intersects the feasible set. That first intersection point will subsequently be the output of an $\ell^1$-minimization algorithm, since all other points with smaller $\ell^1$ norm (namely the points in the earlier $\ell^1$ balls) were separated from the hyperplane $\{\mathbf{x}\} + \ker(\mathbf{A})$. It is apparent that this minimizer might not in general be unique, as if for example the solution space is parallel with the closest face of the $\ell^1$ ball then there would be multiple solutions.

An important visualization to associate with this minimization is that the $\ell^1$ ball is convex, as for $x, y \in B_1$, and $t \in [0, 1]$, we get that

$$\|tx + (1-t)y\|_1 \leq t\|x\|_1 + (1-t)\|y\|_1 \leq t + 1 - t = 1,$$

so $tx + (1-t)y \in B_1$ as well. In addition, its image under a linear map (such as $\mathbf{A}$ in Figure 3) is likewise convex. In particular this convexity is geometrically sharp rather than smooth. As seen in Figure 2, the unit ball in $\mathbb{R}^3$ is a diamond-like solid, with flat faces, and sharp edges and vertices.

Unfortunately, it is not straightforward to generalize this low-dimensional case to the large value of $N$ which we hope to work with in compressed sensing. However, it provides an excellent visualization of a fortunate result, which is that the first intersection point will with high probability be at one of the corners of the $\ell^1$ ball, which are precisely the vectors $\mathbf{z} \in \mathbb{R}^3$ with only one nonzero entry (i.e. those that are 1-sparse).

We may also visualize a similar and related process in the target space, generally $\mathbb{C}^m$, with the images of $\mathbf{x}$ and the $\ell^1$ ball $B_1$. Again adapted from Wright and Ma [19], in Figure 3 we represent the recovery process as expanding the image of the $\ell^1$ ball in $\mathbb{C}^m$ until it hits $\mathbf{A}\mathbf{x}$, where for some $t \in \mathbb{R}$, $\mathbf{A}\mathbf{x} \in \partial(t\mathbf{A}(B_1))$. Here $\partial(t\mathbf{A}(B_1))$
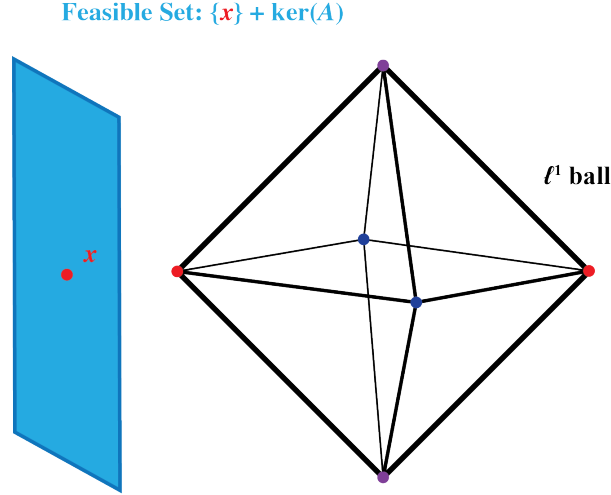
FIGURE 2. Visualization of the $\ell^1$ ball in $\mathbb{R}^3$ ($B_1$) and a feasible set of solutions $\{\mathbf{z} : \mathbf{Az} = \mathbf{Ax}\}$. Performing $\ell^1$-minimization consists of scaling the $\ell^1$ ball by $t$ and taking the first point of intersection of the feasible set with $tB_1$. Because of the jagged nature of $B_1$ this will most likely be a sparse vector. This diagram is based on a figure from Wright and Ma [19].

is the boundary of the image of the $\ell^1$ ball scaled by $t$. Note that Figure 2 and 3 differ in that the feasible set in Figure 2 is visualized by a matrix $\mathbf{A}$ with two-dimensional kernel (thereby producing a plane in $\mathbb{R}^3$, while we do not necessarily have the same restriction on $\mathbf{A}$ for Figure 3.

The key takeaway here is that recovery will be successful if the $\ell^1$ normalized version of $\mathbf{x}$, $\frac{\mathbf{x}}{\|\mathbf{x}\|_1} \in B_1$, gets mapped under $\mathbf{A}$ to the outside of the convex polytope $\mathbf{A}(B_1)$, so if $\mathbf{A}\frac{\mathbf{x}}{\|\mathbf{x}\|_1} \in \partial(\mathbf{A}B_1)$. In particular, note that in this example while the vertices of the convex polytope are the images of the vertices of the unit ball $B_1$ (which in turn are the 1-sparse vectors with only one nonzero entry). As we increase sparsity, and look at the images of the edges of $B_1$ (which are the 2-sparse vectors with two nonzero entries), we see that some, though not all, of these edges get mapped to the outside of the polytope, which is visualized by the line segments in the interior of the polytope. Thus, at least for this 3-dimensional case, once we get past the 1-sparse level, recovery is not guaranteed.

However, when extending this picture to higher dimensions, we will find it actually becomes easier for $\ell^1$-minimization to recover signals as the target space and associated convex polytope grow in dimension. For particular choices of $\mathbf{A}$ (an understanding of which choice we will spend Section 2 exploring), the multi-dimensional faces of the boundary of $\mathbf{A}(B_1)$ will represent the images of signals with a suitably high sparsity, which we will also formalize soon.

In addition, the volume of the $\ell^1$ ball $B_1^N := \{\mathbf{x} \in \mathbb{R}^N : \|\mathbf{x}\|_1 \leq 1\}$ will approach zero as $N \to \infty$, which helps to intuitively understand why sparse recover will grow easier as $N$ increases. While we cannot visualize this progression, the best proxy is to think of the ball as becoming progressively more spiky, with the spiky parts corresponding to sparse vectors, which naturally entails easier signal recovery for such sparse signals.
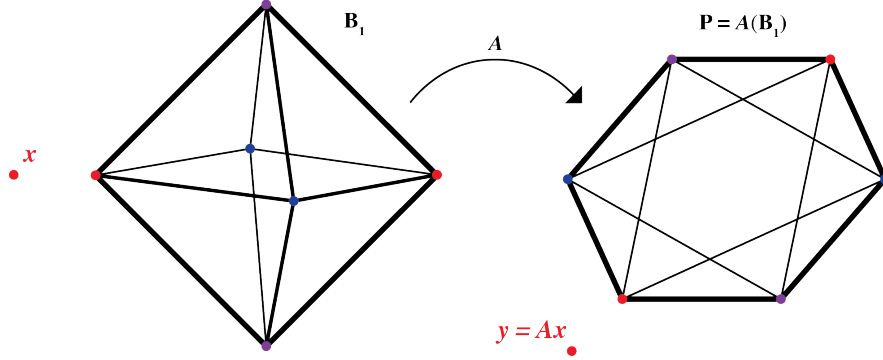
FIGURE 3. Visualization of a possible mapping of the $\ell^1$ ball $B_1$ in $\mathbb{R}^3$ to the convex polytope $P = \mathbf{A}(B_1)$ in $\mathbb{R}^2$. If we scale the sets by $t$, the first point of intersection of $\mathbf{y}$ and $t\mathbf{A}(B_1)$ will correspond to the output of $\ell^1$-minimization. This diagram is based on a figure from Wright and Ma [19].

In particular, the ratio of the volumes of the $\ell^1$ unit ball $B_1^N$ to the $\ell^2$ unit ball $B_2^N := \{\mathbf{x} \in^N \colon \|\mathbf{x}\|_2 \leq 1\}$, using formulas for their volumes found in [18], will approach 0 as $N \to \infty$. Furthermore, if one inscribed a Euclidean ball inside $B_1^N$, the $\ell^1$ ball of dimension $N$, the volume of that Euclidean ball would converge to 0 as $N \to \infty$ [17]. This latter point can be interpreted as showing that the central part of $B_1^N$ becomes very small, with most of its volume present at the tips, hence the notion of a spiky geometry, though we are unable to literally visualize the ball in higher dimensions.

These observations provide for us a much better picture of why $\ell^1$-minimization is in fact a worthy substitute for $\ell^0$-minimization, and as such we are ready to develop an understanding of how to choose cooperative measurement matrices $\mathbf{A}$ to pair with this method.

## 2. THEORETICAL GUARANTEES FOR SUCCESSFUL RECOVERY

This is where the fun begins. In this section we introduce results, usually of the following form: "If the measurement matrix $\mathbf{A} \in \mathbb{C}^{m \times N}$ behaves the following regularity property, then $\mathbf{x} \in \mathbb{C}^N$ is the unique solution to the optimization problem (0.3)". Continuing the trend established in Section 1, we will often justify these regularity properties through a geometric interpretation. Two such results involve the coherence and restricted isometry constants of a matrix, which are developed in Sections 2.1 and 2.2, and Sections 2.3 and 2.4 detail why random matrices are necessary to satisfy the restricted isometry property, a recovery condition based on the restricted isometry constant.

2.1. **Coherence.** First we introduce coherence, a measure associated with a given matrix $\mathbf{A}$ which gives a sense of the linear dependence between the columns of $\mathbf{A}$. In particular, it is easiest to recover a sparse vector supported on $S \subset [N]$ when we know that the columns of $\mathbf{A}$ on that index set are nearly mutually orthogonal, or intuitively, when the columns are more distinguishable from each other. The coherence measure
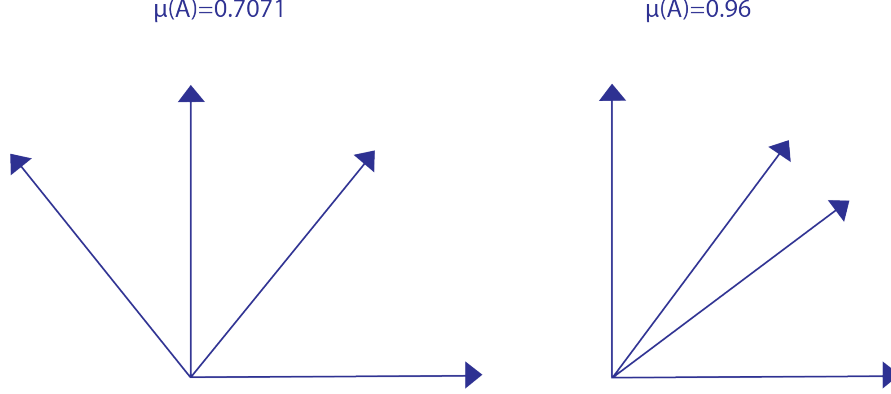
μ(A)=0.7071          μ(A)=0.96

FIGURE 4. Visualization of a general coherence computation in $\mathbb{R}^2$ for two sets of four unit vectors, based on a figure from Wright and Ma [19]. The left-hand set is spaced apart by $\frac{\pi}{4}$ radians each, and the right-hand set consists the two standard unit vectors and the vectors $(\frac{3}{5}, \frac{4}{5})^\top$ and $(\frac{4}{5}, \frac{3}{5})^\top$.

gives a value corresponding to the worst case for any index set of size $s$, so the lower the coherence, the more easily we would expect to achieve sparse recovery. First we define the general coherence function $\mu$ (with implicit argument $\mathbf{a} \in \mathbb{C}^{m \times N}$ where $\mathbf{a}_i \in \mathbb{C}^m$ is the $i$th column vector of $\mathbf{A}$)

$$(2.1) \qquad \mu := \max_{1 \le i \ne j \le N} |\langle \mathbf{a}_i, \mathbf{a}_j \rangle|,$$

which is the worst case scenario for the dependence between any two of the column vectors. Note that $\langle \cdot, \cdot \rangle$ is the standard inner product for $\mathbb{C}^m$, so $|\langle \mathbf{a}_i, \mathbf{a}_j \rangle|$ is interpreted geometrically as the dependence between the two column vectors. A quick visualization in two dimensions of what constitutes a more incoherent set of vectors is given in Figure 4, noting that as the dimensionality $m$ of the column vectors increases, it becomes easier to space out many vectors evenly.

The $\ell^1$-coherence function, which we more commonly use, and is more useful for recovery guarantees, is defined based on some sparsity $s$, and looks at a worst case cumulative dependence which is more important to consider.

**Definition 2.1.** For a matrix $\mathbf{A} \in \mathbb{C}^{m \times N}$ with $\ell^2$-normalized columns $\mathbf{a}_1, ..., \mathbf{a}_N$ with $\|\mathbf{a}_i\| = 1, \forall i \in [N]$, the $\ell^1$ *coherence* function, with argument $s \in [N-1]$, is

$$(2.2) \qquad \mu_1(s) := \max_{i \in [N]} \left[ \max_{S \subset [N] \setminus \{i\}, \text{card}(S) = s} \left\{ \sum_{j \in S} |\langle \mathbf{a}_i, \mathbf{a}_j \rangle| \right\} \right].$$

Note that we define coherence for matrices with $\ell^2$-normalized columns since we can always rescale the entries of the signal vector to obtain the appropriate response vector $\mathbf{y}$. This function measures the worst case scenario for the total dependence between one column of $\mathbf{A}$ and the columns of $\mathbf{A}$ from a disjoint index set of size $s$ [8]. Initially this appears to be a complicated formula, but we can check that this function aligns with our intuitive understanding by seeing that for an orthogonal matrix, the coherence will be 0 for any sparsity $s$. Of course in our case for $m < N$, it is impossible to construct an orthogonal $m \times N$ matrix, so we have to rely on this concept of small worst case dependence between sparse sets and single other vectors.

Coherence ends up being a nice function to work with, and there are plenty of results for bounds on coherence, many of which are covered by Foucart and Rauhut [8]. In particular, for any matrix $\mathbf{A} \in \mathbb{C}^{m \times N}$ the coherence $\mu$ increases as $N$ increases and decreases as $m$ increases, but much faster in the latter case, given by the result that the coherence satisfies [8]

$$(2.3) \qquad \mu \geq \sqrt{\frac{N-m}{m(N-1)}}.$$

In addition there is a relation between the general and $\ell^1$ coherence functions:

$$(2.4) \qquad \mu \leq \mu_1(s) \leq s\mu, \forall s \in [N-1].$$

See Appendix B for a proofs of the results (2.3) and (2.4).

We care specifically about what the coherence of a matrix means for our ability to recover a sparse signal via $\ell^1$-minimization. In fact there is a relatively straightforward condition, which is given in the following theorem [8]:

**Theorem 2.2.** *For $\mathbf{A} \in \mathbb{C}^{m \times N}$ a matrix with $\ell^2$-normalized columns, if*

$$(2.5) \qquad \mu_1(s) + \mu_1(s-1) < 1$$

*then every s-sparse $\mathbf{x} \in \mathbb{C}^N$ is recovered exactly, both by basis pursuit (Section 3.1), and orthogonal matching pursuit (Section 3.2) after at most s iterations.*

Basis pursuit and orthogonal matching pursuit are algorithms for compressed sensing, more detailed explanations for which are given in the associated sections, but for now note that basis pursuit is essentially shorthand for performing the $\ell^1$ minimization (0.3), and orthogonal matching pursuit is a greedy method that iteratively chooses an index set of size $s$, adding one new index to this set at each of its iterations. The following proof sketch is adapted from Foucart and Rauhut [8].

*Proof.* First we sketch the proof for basis pursuit. This proof relies on Theorem A.2, which says that to prove exact recovery via basis pursuit we need to prove the null space property of order $s$:

$$(2.6) \qquad \|\mathbf{v}_S\|_1 < \|\mathbf{v}_{\bar{S}}\|_1,$$

where $S \subset [N]$ has $\text{card}(S) = s$, $\bar{S}$ is the complement of $S$ in $[N]$, and $\mathbf{v} \in \ker(\mathbf{A})$ is nonzero, which means $\mathbf{v} = \sum_{j \in [N]} v_j \mathbf{a}_j$ for $\mathbf{a}_j$ the $j$th column of $\mathbf{A}$. We want to bound $\|\mathbf{v}_S\|_1 = \sum_{i \in S} |v_i|$, so let's look at $v_i$. Since $\mathbf{v} \in \ker(\mathbf{A})$, we know

$$\left\langle \sum_{j=1}^N v_j \mathbf{a}_j, \mathbf{a}_i \right\rangle = 0 \Rightarrow v_i = v_i \langle \mathbf{a}_i, \mathbf{a}_i \rangle = -\sum_{k \in \bar{S}} v_k \langle \mathbf{a}_k, \mathbf{a}_j \rangle - \sum_{j \in S, j \neq i} v_j \langle \mathbf{a}_j, \mathbf{a}_i \rangle.$$

Thus using the triangle inequality we get that

$$|v_i| \leq \sum_{k \in \bar{S}} |v_k| |\langle \mathbf{a}_k, \mathbf{a}_i \rangle| + \sum_{j \in S, j \neq i} |v_j| |\langle \mathbf{a}_j, \mathbf{a}_i \rangle|,$$

after which to get a bound on $\|\mathbf{v}_S\|_1$ we sum over all $i \in S$, getting

$$\|\mathbf{v}_S\|_1 \leq \sum_{k \in \bar{S}} |v_k| \sum_{i \in S} |\langle \mathbf{a}_\ell, \mathbf{a}_i \rangle| + \sum_{j \in S} |v_j| \sum_{i \in S, i \neq j} |\langle \mathbf{a}_j, \mathbf{a}_i \rangle|$$

$$\leq \sum_{k \in \bar{S}} |v_\ell| \mu_1(s) + \sum_{j \in S} |v_j| \mu_1(s-1) = \mu_1(s) \|\mathbf{v}_{\bar{S}}\|_1 + \mu_1(s-1) \|\mathbf{v}_S\|_1.$$

For this last inequality we used the fact that the coherence with argument $s$ is a maximum of $\sum_{i \in S} |\langle \mathbf{a}_k, \mathbf{a}_i \rangle|$ terms, over $\mathrm{card}(S) = s$ and $k \notin S$. We then re-arrange this inequality to obtain

$$\|\mathbf{v}_S\|_1 (1 - \mu_1(s-1)) \leq \|\mathbf{v}_{\bar{S}}\| \mu_1(s),$$

which becomes (2.6) when we apply the strict inequality $\mu_1(s) < 1 - \mu_1(s-1)$.

$\square$

*Proof.* Second, we sketch the proof for orthogonal matching pursuit. For $(\mathbf{a}_j)_{j \in [N]}$ the columns of $\mathbf{A}$, Proposition A.3 states that we need to prove that for any index set $S \subset [N]$ with $\mathrm{card}(S) = s$, that the matrix $\mathbf{A}_S$ of $\mathbf{A}$ on the support set $S$ is injective, and for all $\mathbf{w} = \mathbf{A}\mathbf{x}$ for $s$-sparse $\mathbf{x}$,

$$\max_{j \in S} |\langle \mathbf{w}, \mathbf{a}_j \rangle| > \max_{k \notin S} |\langle \mathbf{w}, \mathbf{a}_k \rangle|.$$

Denoting $\mathbf{w} = \sum_{i=S} w_i \mathbf{a}_i$ for $w_i$ the nonzero coefficients of some $\mathbf{x}$, we see that for $k \notin S$,

$$|\langle \mathbf{w}, \mathbf{a}_k \rangle| = \left| \sum_{i \in S} w_i \langle \mathbf{a}_i, \mathbf{a}_k \rangle \right| \leq \sum_{i \in S} |w_i| |\langle \mathbf{a}_i, \mathbf{a}_k \rangle| \leq |w^*| \mu_1(s),$$

using a simple triangle inequality, with $|w^*| = \max_{i \in S} |w_i|$ which exists since $S$ is a finite set, and since the computation for $\mu_1(s)$ is the max of $\sum_{i \in S} |\langle \mathbf{a}_i, \mathbf{a}_k \rangle|$ where $k \notin S$. Then for any $j \in S$, we have

$$|\langle \mathbf{w}, \mathbf{a}_j \rangle| = \left| \sum_{i \in S} w_i \langle \mathbf{a}_i, \mathbf{a}_k \rangle \right| \geq |w_j| |\langle \mathbf{a}_j, \mathbf{a}_j \rangle| - \sum_{i \in S, i \neq j} |w_i| |\langle \mathbf{a}_i, \mathbf{a}_j \rangle|$$
$$\geq |w_j| - |w^*| \mu_1(s-1),$$

where the first inequality results from the triangle inequality applied to $w_j \langle \mathbf{a}_j, \mathbf{a}_j \rangle = \langle \mathbf{w}, \mathbf{a}_j \rangle - \sum_{i \in S, i \neq j} w_i \langle \mathbf{a}_i, \mathbf{a}_j \rangle$, we use the fact that each $\mathbf{a}_j$ is $\ell^2$ normalized to get that $\langle \mathbf{a}_j, \mathbf{a}_j \rangle = \|\mathbf{a}_j\|_2^2 = 1$, and using the same bound on the coherence, this time for argument $s-1$. The maximum of these such values occurs when $w_j = w^*$, so

$$|\langle \mathbf{w}, \mathbf{a}_j \rangle| \geq |w^*| - |w^*| \mu_1(s-1)$$

for all $j \in S$. Thus, by our assumption, $\mu_1(s) + \mu_1(s-1) < 1$, so conclude that

$$\max_{j \in S} |\langle \mathbf{w}, \mathbf{a}_j \rangle| \geq |w^*| - |w^*| \mu_1(s-1) > |w^*| \mu_1(s) \geq \max_{k \notin S} |\langle \mathbf{w}, \mathbf{a}_k \rangle|,$$

which is what we wanted. The proof concludes with $\mathbf{A}_S$ being injective, which we won't include here but is a result of Corollary 5.4 in [8]. $\square$

Proofs of Theorem A.2 and Prop A.3 are given in Appendix A for the interested reader. This above proof is nice since it showcases why coherence is in many ways a natural metric and delineator between the support set $S$ and its complement $\bar{S}$.

We discuss exact formulations for orthogonal matching pursuit and basis pursuit later in Section 3, but for now it suffices to note that simple conditions such as the above involving coherence allows for recovery for multiple different methods for performing $\ell^1$-minimization. Foucart and Rauhut additionally note that in order to obtain a matrix with low enough coherence to achieve recovery, the minimum number of measurements is on the order of $m \geq Cs^2$ [8].

We obtain this quadratic lower bound in $s$ for $m$ in particular using (2.4). For a matrix with coherence bounded by $\mu \leq c/\sqrt{m}$ (which can be obtained reliably as shown in [8]), we can get $\mu_1(s) + \mu_1(s-1) \leq (2s-1)\mu < 1$ when $m > c^2(2s-1)^2$, which is on the order of $s^2$.

## 2.2. Restricted Isometry Property.
While it may not be immediately obvious, this minimum number of measurements, while certainly a worst-case scenario, is suboptimal, as this worst case value for $m$ scales quadratically with the sparsity $s$, a bound known early in the field as the "quadratic bottleneck" [8]. Overcoming this bottleneck was a major item of research at this time, and we are able to do so through introducing our next notion for the "niceness" of a matrix, called the *restricted isometry property*.

The restricted isometry property relies on the *restricted isometry sonstant*, a measure similar in interpretation to coherence, as it concerns a worst case bound for the action of a matrix $\mathbf{A}$ restricted to an index set $S \subset [N]$. In this case the terminology is suitably descriptive, as the intuition is that matrices with small restricted isometry constant act like isometries (distance-preserving maps) on their restrictions to $s$-sparse vectors. In particular, we define the constant as a function of sparsity $s$ and implicitly of matrix $\mathbf{A}$.

**Definition 2.3.** The *restricted isometry constant $\delta_s$* of a matrix $\mathbf{A} \in \mathbb{C}^{m \times N}$ for sparsity $s$, is defined to be the smallest $\delta \geq 0$ with, for all $s$-sparse $\mathbf{x} \in \mathbb{C}^N$,

$$(2.7) \qquad (1-\delta)\|\mathbf{x}\|_2^2 \leq \|\mathbf{A}\mathbf{x}\|_2^2 \leq (1+\delta)\|\mathbf{x}\|_2^2.$$

Note that depending on the structure of the matrix $\mathbf{A}$, $\delta_s$ could take any value in the positive reals $[0, \infty)$, including $\delta_s > 1$. The value $\delta_s$ has an equivalent formulation in terms of the matrix norm of $\mathbf{A}_S^* \mathbf{A}_S$ (where the $^*$ operator is the conjugate transpose and $\mathbf{A}_S$ is the restriction of $\mathbf{A}$ to index set $S$ with $\mathrm{card}(S) \leq s$) which measures its closeness to the identity and is given formally in Appendix B.1 [8]. The measure's relationship to coherence is formalized in Appendix B.2.

We say that a matrix $\mathbf{A}$ satisfies the *restricted isometry property* if, for large enough $s$, $\delta_s$ is small in some sense. This definition is rather informal, as there is no neat statement of the restricted isometry property, but our takeaway is rather the general idea that we will be able to prove guarantees that rely on a bound on $\delta_s$ based on the size of $s$.

The notion of smallness for $\delta_s$ gives way, as we might hope, to reconstruction guarantees for our algorithms. The following result, which is the main result on which we focus, is very simple and quite beautiful. For basis pursuit (3.1) we have the following theorem, involving the $2s$th restricted isometry constant $\delta_{2s}$.

**Theorem 2.4.** *If the $2s$th restricted isometry constant of the matrix $\mathbf{A} \in \mathbb{C}^{m \times N}$ satisfies*

$$(2.8) \qquad \delta_{2s} < \frac{1}{3},$$

*then every $s$-sparse $\mathbf{x} \in \mathbb{C}^N$ is the unique solution to the $\ell^1$-minimization (0.3), in that it is recovered exactly from the minimization problem*

$$\text{minimize } \|\mathbf{z}\|_1 \quad \text{subject to } \mathbf{A}\mathbf{z} = \mathbf{A}\mathbf{x}.$$

We will give a sketch of the proof of this important theorem, which will draw on results from [8], and will hopefully give a bit of intuition for why the very nice bound of $\delta_{2s} < \frac{1}{3}$ comes up.

*Proof.* As with the proof of Theorem 2.2 in the basis pursuit case, we make use of Prop A.2 and specifically the reformulation of the null space property (A.2) that says we must prove that $\|\mathbf{v}_S\|_1 \leq \frac{1}{2}\|\mathbf{v}\|_1$ for any nonzero $\mathbf{v} \in \ker(\mathbf{A})$ and $S \subset [N]$ with $\text{card}(S) = s$. This proof relies on showing this version of the null space property through a stronger condition,

$$\|\mathbf{v}_S\|_2 \leq \frac{\rho}{2\sqrt{s}}\|\mathbf{v}\|_1, \tag{2.9}$$

on the same inputs as for (A.2), and defining

$$\rho := \frac{2\delta_{2s}}{1 - \delta_{2s}}.$$

We see that (2.9) implies (A.2) since $\|\mathbf{v}_S\|_1 \leq \sqrt{s}\|\mathbf{v}_S\|_2$ from the properties of the $\ell^1$ and $\ell^2$ balls and norms, and $\rho < 1$ as long as $\delta_{2s} < \frac{1}{2}$, which is one of our hypotheses. It is because of this formulation that the upper bound of $\frac{1}{3}$ for $\delta_{2s}$ is given.

The key to this proof, which will not give fully, is to partition the index set $[N]$ into a sequence of sets $S =: S_0, S_1, S_2, \ldots$ with $S_k$ being the $s$ indices in $\overline{S_0 \cup \cdots \cup S_{k-1}}$ with the largest absolute entries of $\mathbf{v}$. Then our goal is to bound $\|\mathbf{v}_S\|_1 = \|\mathbf{v}_{S_0}\|_1$, which we do first through the definition of $\delta_s$ (2.7), and the fact that $\delta_{2s} \leq \delta_s$:

$$\|\mathbf{v}_{S_0}\|_2^2 \leq \frac{1}{1 - \delta_{2s}}\|\mathbf{A}\mathbf{v}_{S_0}\|_2^2 = \frac{1}{1 - \delta_{2s}}\langle \mathbf{A}\mathbf{v}_{S_0}, -\mathbf{A}\mathbf{v}_{S_1} - \mathbf{A}\mathbf{v}_{S_2} - \cdots \rangle$$

$$= \frac{1}{1 - \delta_{2s}}\sum_{k \geq 1}\langle \mathbf{A}\mathbf{v}_{S_0}, \mathbf{A}(-\mathbf{v}_{S_k})\rangle.$$

Combined with the fact (from Corollary 6.3 in [8] which we give without proof) that $\langle \mathbf{A}(\mathbf{v}_{S_0}), \mathbf{A}(-\mathbf{v}_{S_k})\rangle \leq \delta_{2s}\|\mathbf{v}_{S_0}\|_2\|\mathbf{v}_{S_k}\|_2$, and after dividing by $\|\mathbf{v}_{S_0}\|_2$, we get:

$$\|\mathbf{v}_{S_0}\|_2 \leq \frac{\rho}{2}\sum_{k \geq 1}\|\mathbf{v}_{S_k}\|_2 \leq \frac{\rho}{2\sqrt{s}}\sum_{k \geq 1}\|\mathbf{v}_{S_{k-1}}\|_1 \leq \frac{\rho}{2\sqrt{s}}\|\mathbf{v}_{S_0}\|_1,$$

where we used without proof Lemma 6.10 from [8], which says that

$$\|\mathbf{v}_{S_k}\|_2 \leq \frac{1}{\sqrt{s}}\|\mathbf{v}_{S_{k-1}}\|_1.$$

At this point we are done.

$\square$

For more complete proofs of Corollary 6.3 and Lemma 6.10 from [8] we refer the reader to Foucart and Rauhut, but to give a bit of intuition, Corollary 6.3 follows from the properties of the restricted isometry constant on the set $S_0 \cup S_k$ of size $2s$ (hence why the restricted isometry constant $\delta_{2s}$ shows up). Lemma 6.10 has a geometric proof based on the fact that each absolute entry of $\mathbf{v}_{S_{k-1}}$ is greater than each absolute entry of $\mathbf{v}_{S_k}$.

This condition sounds almost too good to be true, but it is more so just a clean result which is easy to remember; it is in fact possible to make the upper bound larger and thus more optimal [8]. Nevertheless, it remains that using the restricted isometry property in basis pursuit has its limitations, among which is the fact that adding or otherwise shuffling and scaling the measurements can interfere with the restricted isometry constant. In particular, adding a measurement can increase the constant, though we might intuitively guess that adding a measurement would increase the chance of recovery [8].

2.3. **Recovery with Random Matrices.** As mentioned, the quadratic bottleneck unfortunately comes up when framing a matrix in terms of coherence, but it turns out that we can beat this limitation through the restricted isometry constant with the following minimum number of measurements

$$m \geq C_\delta s \ln(eN/s),$$

which will guarantee a restricted isometry constant $\delta_s \leq \delta$ in a random setting which we will define. Here $C_\delta$ is a constant depending on a chosen constant $\delta$, our intended upper bound on the restricted isometry constant, and $e$ is the standard notation for Euler's constant $e \approx 2.718$, and is present as a shorthand for saying $m$ is bounded from below by an $s + s \ln(N/s)$ term. As it turns out, proving that certain deterministic matrices meet the requirements to have this optimal lower bound for the number of measurements is a difficult task, with research into this task currently ongoing [8]. As such, the breakthrough in compressed sensing came with the advent of using random measurement matrices.

Specifically, the most common random matrices used are Gaussian random matrices. Gaussian random matrices are defined as matrices $\mathbf{A} \in \mathbb{R}^{m \times N}$ with each entry an independent standard Gaussian $\mathcal{N}(0, 1)$, which is a random variable with mean 0 and variance 1. The Gaussian distribution is also known as the normal distribution. The probability density function (PDF) of the Gaussian distribution is

$$f(x) = \frac{1}{\sqrt{2\pi}} \exp(-\frac{x^2}{2}),$$

and the cumulative distribution function (CDF) for $X \sim \mathcal{N}(0, 1)$ is $F(x) = \mathbb{P}(X \leq x) = \int_{-\infty}^{x} f(x)dx$ which is usually notated as $\Phi(x)$ since the partial integral does not have a closed form. The PDF is symmetric about the origin, with heavy tails, and as such the Gaussian PDF is often referenced as a "bell curve". See Figure 9 in Section 3.5 for a plot of this PDF.

Commonly, the $\ell^2$-normalized version of such a matrix $\frac{1}{\sqrt{m}}\mathbf{A}$ is taken, so for any column $\mathbf{a}_i$ of $\mathbf{A}$,

$$\mathbb{E}(\|\frac{1}{\sqrt{m}}\mathbf{a}_i\|_2^2) = \frac{1}{m} \sum_{j=1}^{m} \mathbb{E}(\mathbf{a}_{ij}^2) = 1,$$

since each $\mathbf{a}_{ij} \sim \mathcal{N}(0, 1)$ independently so $\mathbb{E}(\mathbf{a}_{ij}^2) = \text{var}(\mathbf{a}_{ij}) + \mathbb{E}(\mathbf{a}_{ij})^2 = 1 + 0 = 1$. Note that we have that $\frac{1}{\sqrt{m}}\mathbf{A}$ is $\ell^2$-normalized in the sense that the *square* of the $\ell^2$ norm of each column is 1 in expectation. From Jensen's inequality, which says that for convex $g : \mathbb{R} \to \mathbb{R}$, $g(E(X)) \leq E(g(X))$, we know that $\mathbb{E}(\|\frac{1}{\sqrt{m}}\mathbf{a}_i\|_2)^2 \leq \mathbb{E}(\|\frac{1}{\sqrt{m}}\mathbf{a}_i\|_2^2)$, but scaling by $\frac{1}{\sqrt{m}}$ is easy and turns out to be good enough, as we will see in Section 2.4.

Another distribution over which measurement matrices are taken is often called the Bernoulli random matrix, which counter-intuitively does not have Bernoulli entries but rather independent Rademacher entries. Thus the entries take 1 or $-1$ with equal probability $\frac{1}{2}$ (the confusion is since Bernoulli random *variables* are usually given as taking the values of 1 or 0 with equal probability $\frac{1}{2}$). To minimize confusion we will simply call such matrices Rademacher random matrices, to emphasize that their

entries are independent Rademacher random variables, which have PMF

$$f(x) = \begin{cases} \frac{1}{2} & x = 1 \\ \frac{1}{2} & x = -1 \\ 0 & \text{otherwise.} \end{cases}$$

More generally, compressed sensing works with a class of random matrices defined as subgaussian, a class in which both Gaussian and Rademacher random matrices fall [8]. This terminology is based on the class of subgaussian random variables, which include Gaussian and Rademacher random variables.

**Definition 2.5.** A random variable $X$ is *subgaussian* if there are constants $\beta, \kappa > 0$ with

$$(2.10) \qquad \mathbb{P}(|X| \geq t) \leq \beta e^{-\kappa t^2} \quad \forall t > 0.$$

The left hand side quantity we are bounding is the often called the reverse-CDF of $X$ (with reference to the CDF $F(t) = \mathbb{P}(X \leq t)$), and we can think of subgaussian random variables as those for which the log of their reverse CDF is quadratic in $t$ in the sense that it is bounded above by $-t^2$ up to a constant factor. Examples of subgaussian random variables include the Gaussian, which is subgaussian with parameters $\beta = 1, \kappa = 1/2$, as well as Rademacher random variables and all almost surely bounded random variables. These variables are classified together in this way due to the behavior of their tails, which allow them to satisfy many nice properties. They contrast with subexponential random variables, which have fatter tails, a la the exponential distribution. Note that while subexponentials have heavier tails, both subgaussians and subexpontentials broadly speaking have light tails since their reverse CDFs approach zero exponentially in the positive direction. See Section 3.5 for a comparison of compressed sensing using subgaussian versus subexponential matrices. The definition of subgaussian random variables is naturally extended to random matrices in the same way as we defined for Gaussian and Rademacher matrices.

**Definition 2.6.** A matrix $\mathbf{A} \in \mathbb{R}^{m \times N}$ is a *subgaussian* random matrix if it has independent subgaussian entries with mean 0 and the same subgaussian parameters $\beta, \kappa$.

Important to note is that the entries of such a subgaussian $\mathbf{A}$ are all independent and identically distributed, with the same parameters. Specifying the subgaussian parameters $\beta, \kappa$ is the most common way to define a general subgaussian distribution [8].

With this set of random matrices in mind, we see a path to the achievement of the optimal restricted isometry property through the following theorem.

**Theorem 2.7.** *For $\mathbf{A} \in \mathbb{R}^{m \times N}$ a subgaussian matrix, there exists a constant $C$ depending only on the parameters of the subgaussian distribution $(\beta, \kappa)$ where the restricted isometry constant of $\frac{1}{\sqrt{m}}\mathbf{A}$ satisfies $\delta_s \leq \delta$ with probability at least $1 - \varepsilon$, as long as*

$$(2.11) \qquad m \geq C\delta^{-2}(s\ln(eN/s) + \ln(2\varepsilon^{-1})).$$

Theorem 2.7 provides a criterion for ensuring that the $s$th restricted isometry constant $\delta_s$ of the matrix $\frac{1}{\sqrt{m}}\mathbf{A}$ is below a certain desired threshold $\delta$ with high probability. Theorem 2.4 tells us that bounding the restricted isometry constant provides for sparse recovery, so finding this criterion is an essential step toward recovery with

random matrices that we will complete in the following theorem, which corresponds to a recovery condition in the $\ell^1$-minimization problem (0.3).

**Theorem 2.8.** *For* $\mathbf{A} \in \mathbb{R}^{m \times N}$ *a subgaussian matrix, and* $\varepsilon \in (0, 1)$*, there exist parameters* $C_1, C_2 > 0$ *depending on the subgaussian parameters* $(\beta, \kappa)$ *such that if*

$$(2.12) \qquad m \geq C_1 s \ln(eN/s) + C_2 \ln(2\varepsilon^{-1}),$$

*then with probability at least* $1 - \varepsilon$*, any* $s$*-sparse* $\mathbf{x} \in \mathbb{R}^N$ *is the unique solution to the* $\ell^1$*-minimization (0.3).*

The proof of Theorem 2.8 follows from applying Theorem 2.7 and Theorem 2.4, which combine to remove the $\delta$ parameter in (2.12). The proof of Theorem 2.7 is more engaged, involving numerous intermediate lemmas and theorems in [8], which the reader can consult for the complete picture. We will prove a simpler version of these results in Section 2.4 for Gaussian matrices, which will more clearly show why each term of the lower bound for $m$ is required.

Note that this recovery is first and foremost probabilistic, and the higher we want the probability of recovery the larger we must make $m$ (via decreasing $\varepsilon$). In addition, note that the use of two constants $C_1, C_2$ as opposed to just $C$ in Theorem 2.7 conveys fundamentally the same condition, but we can make the minimum number of measurements smaller with the two constants, which is more helpful in practice.

These recovery conditions take the form of *uniform* recovery, which is when all $s$-sparse vectors $\mathbf{x}$ are recoverable from $\mathbf{A}$ with high probability, whereas *non-uniform* recovery concerns the probability that any given $s$-sparse vector can be recovered. This weaker condition is often easier to achieve, so it has its place in the literature, though we will not cover it any further [8].

2.4. **Proof of the Restricted Isometry Property for Gaussian Matrices.** Concluding the satisfaction of the restricted isometry property for random matrices as a black box is somewhat unsatisfying, especially since the development is such an important part of the start of compressed sensing as a field. The proof relies on various concentration inequalities for subgaussian distributions, which are somewhat advanced probabilistic tools, and the proof is additionally involved as the subgaussian is general, with two constants $\beta$ and $\kappa$ [8]. The reader may refer to Foucart and Rauhut's textbook *A Mathematical Introduction to Compressed Sensing* [8] for a more complete proof of Theorems 2.7 and 2.8, which as mentioned primarily makes use of concentration inequalities on the tails of subgaussian distributions.

However, the standard Gaussian distribution is very well-known and understood, and a proof of the restricted isometry property for a Gaussian matrix requires only basic facts about the Gaussian distribution and the Chernoff bound, whic hare standard results in introductory probability. Thus we will include a proof of Theorem 2.7 specifically for Gaussian random matrices that will paint an illuminating picture of the nature of the restricted isometry property.

**Theorem 2.9.** *The matrix* $\mathbf{A} \in \mathbb{R}^{m \times N}$*, a normalized Gaussian matrix, with i.i.d.* $\mathbf{A}_{ij} \sim \mathcal{N}(0, \frac{1}{m})$ *entries, satisfies the restricted isometry property with restricted isometry constant* $\delta_s \leq \delta$ *with probability* $1 - \varepsilon$ *if*

$$(2.13) \qquad m \geq \frac{2s \log(eN/s) + 2 \log(2\varepsilon^{-1})}{\delta - \log(\delta + 1)}.$$

For this proof we draw on [8, 14], as well as basic results from introductory probability theory.

*Proof.* In order to first compute the restricted isometry constant $\delta_s(\mathbf{A})$, we need to find the smallest $\delta'$ such that for all $s$-sparse $\mathbf{x}$,

$$(2.14) \qquad (1 - \delta')\|\mathbf{x}\|_2^2 \leq \|\mathbf{A}\mathbf{x}\|_2^2 \leq (1 + \delta')\|\mathbf{x}\|_2^2.$$

Here everything is held constant except for $\mathbf{A}$, which is random in the way given, meaning $\|\mathbf{A}\mathbf{x}\|_2^2$ is a random variable taking values $\mathbb{R}$. In particular,

$$\|\mathbf{A}\mathbf{x}\|_2^2 = \sum_{i=1}^{m} |\langle \mathbf{a}_i, \mathbf{x} \rangle|^2,$$

where for $\mathbf{a}_i$ the $i$th row of $\mathbf{A}$, the inner product (which is just the dot product in this real setting) is $\langle \mathbf{a}_i, \mathbf{x} \rangle = \sum_{j=1}^{N} \mathbf{a}_{ij}\mathbf{x}_j \sim \mathcal{N}(0, \frac{\|\mathbf{x}\|_2^2}{m})$ using the standard properties of a linear combination of Gaussian variables, where $\|\mathbf{x}\|_2^2 = \sum_{j=1}^{N} \mathbf{x}_j^2$. Then $\|\mathbf{A}\mathbf{x}\|_2^2$ is a sum of $m$ squared Gaussians, meaning $\|\mathbf{A}\mathbf{x}\|_2^2 \sim \frac{\|\mathbf{x}\|_2^2}{m} \cdot \chi_m^2$ for $\chi_m^2$ a standard $\chi^2$ (Chi-squared) random variable with $m$ degrees of freedom. Then if we want to bound the probability that (2.14) holds for $\delta$, we want to bound $\mathbb{P}(|\|\mathbf{A}\mathbf{x}\|_2^2 - \|\mathbf{x}\|_2^2| \leq \delta\|\mathbf{x}\|_2^2)$ from below. From the distribution we derived for $\|\mathbf{A}\mathbf{x}\|_2^2$ we know that

$$\mathbb{E}\left[\|\mathbf{A}\mathbf{x}\|_2^2\right] = \frac{\|\mathbf{x}\|_2^2}{m} \cdot m = \|\mathbf{x}\|_2^2,$$

since the $\chi_m^2$ expectation is $m$ [14], so bounding the complement of this probability from above is a natural strategy using basic concentration bounds from probability theory. Using the Chernoff bound (which is a standard probabilistic tool [14]) with a generic $t > 0$ on random variable $\|\mathbf{A}\mathbf{x}\|_2^2$ and constant $\|\mathbf{x}\|_2^2(1 + \delta)$, we obtain an upper bound of

$$\mathbb{P}(\|\mathbf{A}\mathbf{x}\|_2^2 \geq \|\mathbf{x}\|_2^2(1 + \delta)) \leq \frac{\mathbb{E}\left[\exp(t\|\mathbf{A}\mathbf{x}\|_2^2)\right]}{\exp(t\|\mathbf{x}\|_2^2(1 + \delta))} = \frac{\left(1 - \frac{2t\|\mathbf{x}\|_2^2}{m}\right)^{-m/2}}{\exp(t\|\mathbf{x}\|_2^2(1 + \delta))},$$

which is a result of the moment generating function (defined as $M(t) = \mathbb{E}\left[\exp(tX)\right]$ for random variable $\mathbf{X}$) of the $\chi_m^2$ distribution [14]. In the same vein as the standard probability trick, we can minimize this upper bound as a function of $t$, since this is true for all $t > 0$. We omit the computation, which is a matter of simple calculus, and yields

$$t = \frac{m}{2\|\mathbf{x}\|_2^2}\left(\frac{\delta}{1 + \delta}\right).$$

Upon substituting this value of $t$, we retrieve the bound

$$(2.15) \qquad \mathbb{P}(\|\mathbf{A}\mathbf{x}\|_2^2 \geq \|\mathbf{x}\|_2^2(1 + \delta)) \leq \frac{\left(1 - \frac{\delta}{1+\delta}\right)^{-m/2}}{\exp(-m\delta/2)} = \exp\left(-m\delta/2\right)(1 + \delta)^{m/2},$$

which is the upper part of our bound and the first half of $\mathbb{P}(|\|\mathbf{A}\mathbf{x}\|_2^2 - \|\mathbf{x}\|_2^2 \geq \delta\|\mathbf{x}\|_2^2)$. Now we want to upper bound the lower probability $\mathbb{P}(\|\mathbf{A}\|_2^2 \leq \|\mathbf{x}\|_2^2(1 - \delta))$. This formulation is not equipped for the same form of the Chernoff bound used earlier,

but taking the negation before exponentiation gives an alternative formulation of the Chernoff bound for a random variable $X$:

$$\mathbb{P}(|\mathbf{X}| \leq a) \leq \frac{\mathbb{E}\left[\exp(-t|\mathbf{x}|)\right]}{\exp(-ta)}.$$

Using this formulation, we obtain the bound

$$\mathbb{P}(\|\mathbf{A}\mathbf{x}\|_2^2 \leq \|\mathbf{x}\|_2^2(1-\delta)) \leq \frac{\left(1 + \frac{2t\|\mathbf{x}\|_2^2}{m}\right)^{-m/2}}{\exp(-ts(1-\delta))},$$

since we are essentially just substituting $-t$ for $t$ and $(1-\delta)$ for $(1+\delta)$, meaning the moment generating function computation still works, and furthermore the minimizing value of $t$ is

$$t = -\frac{m}{2\|\mathbf{x}\|_2^2}\left(\frac{\delta}{1-\delta}\right),$$

meaning we have after substitution:

(2.16) $$\mathbb{P}(\|\mathbf{A}\mathbf{x}\|_2^2 \leq \|\mathbf{x}\|_2^2(1-\delta)) \leq \exp(m\delta/2)(1-\delta)^{m/2}.$$

Combining (2.15) and (2.16) we obtain

$$\mathbb{P}(|\|\mathbf{A}\mathbf{x}\|_2^2 - \|\mathbf{x}\|_2^2 \geq \delta\|\mathbf{x}\|_2^2) \leq \exp\left(-m\delta/2\right)(1+\delta)^{m/2} + \exp(m\delta/2)(1-\delta)^{m/2}$$
$$\leq 2\left(-m\delta/2\right)(1+\delta)^{m/2},$$

since the numerical value of the former upper bound is the higher value [14].

This is the probability bound for a single $\mathbf{x}$, but in order to satisfy the restricted isometry property with $\delta_s \leq \delta$, we need this bound to be true for all $s$-sparse $\mathbf{x}$, of which there are $\binom{N}{s}$, so by a union bound, we obtain

$$\mathbb{P}(|\|\mathbf{A}\mathbf{x}\|_2^2 - \|\mathbf{x}\|_2^2 \geq \delta\|\mathbf{x}\|_2^2, \forall \mathbf{x} \in V_s) \leq 2\binom{N}{s}\left(-m\delta/2\right)(1+\delta)^{m/2}$$
$$\leq 2\left(\frac{eN}{s}\right)^s \left(-m\delta/2\right)(1+\delta)^{m/2},$$

using the classic bound for a binomial coefficient $\binom{n}{k} \leq \left(\frac{en}{k}\right)^k$ [14].

We want this upper bound to be $\leq \varepsilon$ in order to conclude that $\mathbf{A}$ satisfies $\delta_s \leq \delta$ with probability at least $1 - \varepsilon$, so we solve the following inequality for $m$:

$$2\exp(-\frac{m}{2}(\delta - \log(\delta + 1)) + s\log(eN/s)) \leq \varepsilon.$$

After taking the log of each side, it becomes a linear inequality in $m$ which is simple to solve, and gives us exactly the inequality (2.13) which was assumed so we indeed conclude that $\delta_s \leq \delta$ with probability $\geq 1 - \varepsilon$.

$\square$

Hopefully this proof gives a good intuition for why Gaussian variables in particular are good candidates for satisfying the restricted isometry property, specifically as a result of their short tails. However, note that other short-tailed distributions (i.e. those whose reverse CDFs decay exponentially) do not produce the same results as Gaussians or subgaussians, notably the Exponential and corresponding subexponential family [8]. To visualize the difference in sparse recovery performance between random matrices of each type, see Section 3.5 for a brief simulated comparison.

## 3. Algorithms for Sparse Recovery

With a field such as compressed sensing, where much of its value lies in performing a recovery with as few measurements as possible, we naturally care about the ease of implementation in practice with real data, and thus the field is entwined with algorithmic design. We've already seen in Section 2 that recovery guarantees are usually framed in terms of their corresponding algorithms, including basis pursuit and orthogonal matching pursuit.

In this section we detail the various types of algorithms which have been used in the field, which fall into three major categories: convex optimization (Section 3.1), greedy algorithms (Section 3.2), and thresholding algorithms (Section 3.3). We will also compare via simulation the performance of these algorithms based on the number of measurements and sparsity of the problem in Section 3.4 and compare how the choice of measurement matrix impacts recovery rate in Section 3.5.

3.1. **Convex Optimization via Basis Pursuit.** The first and most intuitive algorithm we give is basis pursuit. Below is the pseudocode for basis pursuit (BP), which is actually somewhat useless because it is actually just saying to perform the $\ell^1$-minimization problem (0.3).

---

**Algorithm 1:** Basis Pursuit

*Input:* measurement matrix $\mathbf{A}$, response vector $\mathbf{y}$;
*Do:* $\mathbf{x}^* = \arg\min ||\mathbf{z}||_1$ with $\mathbf{y} = \mathbf{A}\mathbf{z}$;
*Output:* The vector $\mathbf{x}^*$.

---

Here argmin entails choosing the vector $\mathbf{z} \in \mathbb{C}^N$ with $\mathbf{y} = \mathbf{A}\mathbf{z}$ that has a minimal $\ell^1$ norm. Basis pursuit is no more than the $\ell^1$-minimization problem (0.3), but its implementation is the more interesting question, which involves algorithms like linear programming, which constitute, for a set of problems with linear constraints, many well-documented ways of computing a solution. In Appendix C we describe how to formulate basis pursuit via linear programming and a general intuition for how the computation method works.

3.2. **Greedy Solutions via Orthogonal Matching Pursuit.** The following is the pseudocode for orthogonal matching pursuit (OMP), which iteratively updates running counters for the candidate signal $\mathbf{x}^n$ and support set for $\mathbf{x}^n$, $S^n$. Note that $\mathbf{A}^*$ is notation for the adjoint matrix, or conjugate transpose, of $\mathbf{A}$.

---

**Algorithm 2:** Orthogonal Matching Pursuit

*Input:* measurement matrix $\mathbf{A}$, response vector $\mathbf{y}$, max sparsity $\bar{n}$;
*Initialize:* $\mathbf{x}^0 = \mathbf{0}, S^0 = \varnothing$;
*Iterate:* until $n = \bar{n}$;
$$S^{n+1} := S^n \cup \{j_{n+1}\}, j_{n+1} := \arg\max_{j \in [N]}\{|\mathbf{A}^*(\mathbf{y} - \mathbf{A}\mathbf{x}^n)_j|\}$$
$$\mathbf{x}^{n+1} := \arg\min_{\text{supp}(\mathbf{z}) \subset S^{n+1}}\{||\mathbf{y} - \mathbf{A}\mathbf{z}||_2\};$$
*Output:* $\mathbf{x}^* = \mathbf{x}^{\bar{n}}$.

---

The input $\bar{n}$ is the maximum number of iterations we will allow the algorithm to run, and is usually set to our target sparsity $s$ if it is known.

Parallel to the function of argmin for basis pursuit, here argmax entails choosing the index $j \in [N]$ with the maximal value of $|\mathbf{A}^*(\mathbf{y} - \mathbf{A}\mathbf{x}^n)_j|$, where $\mathbf{A}^*(y - \mathbf{A} - \mathbf{A}\mathbf{x}^n)_j$ is the $j$th element of the matrix $\mathbf{A}^*(y - \mathbf{A} - \mathbf{A}\mathbf{x}^n)$. Unlike basis pursuit, which is a simple restatement of $\ell^1$-minimization, there is more explanation required to reason why orthogonal matching pursuit should at least be a suitable proxy for $\ell^1$-minimization. The following lemma and associated proof help to explain why the index added at each step is optimal.

**Lemma 3.1.** *For $\mathbf{A} \in \mathbb{C}^{m \times N}$ a matrix with $\ell^2$-normalized columns, and given $S \subset [N]$, $\mathbf{v}$ with $\mathrm{supp}(\mathbf{v}) = S$, and $j \in [N]$, if*

$$(3.1) \qquad \mathbf{w} := \mathrm{argmin}_{\mathrm{supp}(\mathbf{z}) \subset S \cup \{j\}}\{||\mathbf{A}\mathbf{z} - \mathbf{y}||_2\}$$

*then*

$$(3.2) \qquad ||\mathbf{A}\mathbf{w} - \mathbf{y}||_2^2 \leq ||\mathbf{A}\mathbf{v} - \mathbf{y}||_2^2 - |(\mathbf{A}^*(\mathbf{y} - \mathbf{A}\mathbf{v}))_j|^2.$$

Treating $\mathbf{x}^n$ as $\mathbf{v}$, and $\mathbf{x}^{n+1}$ as $\mathbf{w}$, we see that Lemma 3.1 tells us that choosing $j_{n+1}$ to maximize $|(\mathbf{A}^*(\mathbf{y} - \mathbf{A}\mathbf{v}))_j|^2$ will be optimal in minimizing the upper bound on $||\mathbf{A}\mathbf{w} - \mathbf{y}||_2$. At each iteration, the algorithm for orthogonal matching pursuit chooses a new index $j_{n+1}$ to add to the index set $S^n$ to decrease $||\mathbf{A}\mathbf{x}^n - \mathbf{y}||_2$ by as much as possible.

*Proof.* In computing (3.1), we see that any $\mathbf{z} = \mathbf{v} + t\mathbf{e}_j$ is supported on $S \cup \{j\}$, so $||\mathbf{A}\mathbf{w} - \mathbf{y}||_2 \leq \min_{t \in \mathbb{C}} ||\mathbf{A}(\mathbf{v} + t\mathbf{e}_j) - \mathbf{y}||_2$, at which point we can further bound the left side by taking $t = \rho e^{i\theta}$ as a generic complex number:

$$||\mathbf{A}\mathbf{v} + t\mathbf{A}\mathbf{e}_j - \mathbf{y}||_2^2 = ||\mathbf{A}\mathbf{v} - \mathbf{y}||_2^2 + ||t\mathbf{A}\mathbf{e}_j||_2^2 - \mathrm{Re}(\bar{t}\langle \mathbf{A}\mathbf{e}_j, \mathbf{A}\mathbf{v} - \mathbf{y}\rangle)$$
$$= ||\mathbf{A}\mathbf{v} - \mathbf{y}||_2^2 + \rho^2 - 2\,\mathrm{Re}(\rho e^{-i\theta}(\mathbf{A}^*(\mathbf{y} - \mathbf{A}\mathbf{v}))_j)$$
$$\geq ||\mathbf{A}\mathbf{v} - \mathbf{y}||_2^2 + \rho^2 - 2\rho|(\mathbf{A}^*(\mathbf{y} - \mathbf{A}\mathbf{v}))_j|,$$

from the fact that $||\mathbf{A}\mathbf{e}_j||_2 = 1$ since the columns of $\mathbf{A}$ are $\ell^2$-normalized, and where the closing inequality comes from the fact that the real part of a complex vector is no greater than its magnitude. The optimal bound then results from choosing $t$ to minimize this expression, which is a quadratic polynomial in $\rho$, and completing the square gives the minimizing value of $\rho$ as $|(\mathbf{A}^*(\mathbf{y} - \mathbf{A}\mathbf{v}))_j|$, producing our final, intended inequality

$$||\mathbf{A}\mathbf{w} - \mathbf{y}||_2^2 \leq \min_{t \in \mathbb{C}} ||\mathbf{A}(\mathbf{v} + t\mathbf{e}_j) - \mathbf{y}||_2^2 = ||\mathbf{A}\mathbf{v} - \mathbf{y}||_2 - |(\mathbf{A}^*(\mathbf{y} - \mathbf{A}\mathbf{v}))_j|^2.$$

$\square$

In summary, orthogonal matching pursuit at each step chooses a new index $j_{n+1}$ which will decrease the $\ell^2$ norm distance between the response measurement $\mathbf{y}$ and the image under $\mathbf{A}$ of the candidate signal $\mathbf{x}^n$, and then the new candidate signal $\mathbf{x}^{n+1}$ is chosen on the new support set by minimizing this $\ell^2$ norm difference. It is in this sense that orthogonal matching pursuit is "greedy", since at each iteration in the algorithm it chooses a new index that maximizes its utility at the present step, without regard to any long term behavior, though we will see that orthogonal matching pursuit does turn out to perform rather well in general.

The benefits and downsides of orthogonal matching pursuit in comparison to basis pursuit are somewhat intuitive: because of its greedy nature and use of the $\ell^2$ norm orthogonal matching pursuit is faster and easier to implement than basis pursuit, but it is able to correctly recover a given signal less often than basis pursuit, that

is to say in fewer situations. Nevertheless, orthogonal matching turns out to be a suitable replacement for basis pursuit, with similar enough recovery that its improvements in run time outpace its negatives. Kunis and Rauhut obtained this result for sparse recovery specifically for the data structure of trigonometric polynomials in 2008 [10], and Tropp and Gilbert in 2007 similarly obtained a performance for orthogonal matching pursuit which was close to basis pursuit for the general experimental setting [16]. As perhaps the two most common approaches for compressed sensing, we will give a comparison of their performances on a generated dataset in section 3.4, in a fashion similar to Tropp and Gilbert's simulations.

While orthogonal matching pursuit is the most common greedy algorithm, others exist, such as compressive sampling matching pursuit (CoSaMP) [8], as well as hybrid methods combining greedy methods with convex optimization [13]. Next we will briefly overview a final major category of compressed sensing algorithms.

3.3. **Thresholding Methods for Sparse Recovery.** Basis pursuit involves literally performing the $\ell^1$-minimization problem (0.3) and orthogonal matching pursuit is an intuitive greedy solution for recovering sparse signals, but there is a third major category of algorithms used in compressed sensing, known as thresholding algorithms. The method broadly speaking seeks to achieve the task of sparse recovery, which is essentially an inverse operation, by using the adjoint matrix $\mathbf{A}^*$ [8].

We will not delve too far into thresholding algorithms, but we will give some pseudocode for the most basic form of thresholding to provide some intuition for how they operate. First we define the function $L_s(\mathbf{z})$ being the set of $s$ indices $i$ with the largest values of $|\mathbf{z}_i|$ for $\mathbf{z} \in \mathbb{C}^N$. This function is relevant as thresholding effectively first chooses the support on which to look for solutions and then optimizes within sparse vectors on that support [8]. So a basic thresholding algorithm looks like the following:

---

**Algorithm 3:** Basic Thresholding

*Input:* measurement matrix $\mathbf{A}$, response vector $\mathbf{y}$, sparsity $s$;
*Do:*

$$S^* = L_s(\mathbf{A}^*\mathbf{y})$$

$$\mathbf{x}^* = \text{argmin}_{\mathbf{z}\in\mathbb{C}^N}\{\|\mathbf{y} - \mathbf{A}\mathbf{z}\|_2, \text{supp}(\mathbf{z}) \subset S^*\}.$$

*Output:* $\mathbf{x}^*$.

---

From the simple pseudocode it is clear that for a vector $\text{supp}(\mathbf{x}) = S$ to be recovered, it is necessary for each entry of $\mathbf{A}^*\mathbf{y}$ in this support set to have greater absolute value than every other entry, i.e.

$$\min_{j\in S}|(\mathbf{A}^*\mathbf{y})_j| > \max_{k\notin S}|(\mathbf{A}^*\mathbf{y})_k|.$$

Under this condition, the set $S^*$ from the pseudocode will coincide with $S$ and the optimization will produce the correct $\mathbf{x}^*$.

There are also iterative formulations of this method where at each $n$th step a new support set $S^n$ is chosen to optimize over [8]. The key here with thresholding as compared with orthogonal matching pursuit is that thresholding chooses a set of size
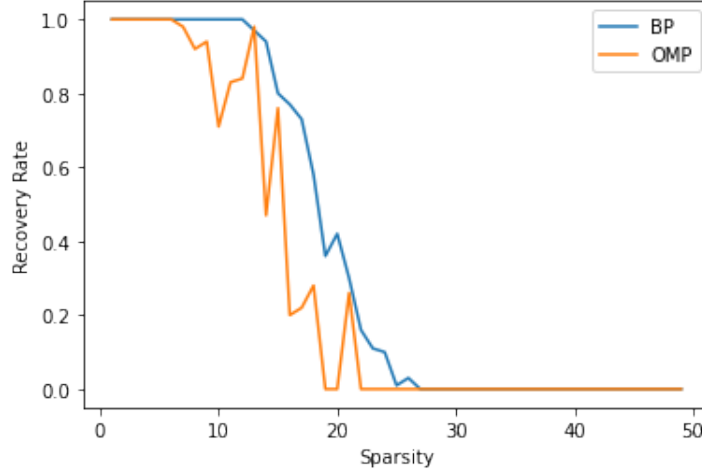
FIGURE 5. Recovery rate of random sparse vectors over a set of 100 Gaussian matrices, as a function of sparsity for fixed ($m = 50, N = 200$).

$s$ all at once (even if it does so multiple times), whereas orthogonal matching pursuit builds a set of size $s$ one index at a time.

Although we will not include thresholding in our simulated comparison of basis pursuit and orthogonal matching pursuit, it remains an important piece of the field of compressed sensing.

3.4. **A Simulated Comparison of Basis Pursuit and Orthogonal Matching Pursuit.** In this subsection we use Python to code implementations of basis pursuit and orthogonal matching pursuit, and perform various comparisons of the two algorithms, varying the sparsity $s$, parameter dimension $N$, and number of measurements $m$. The Python functions used were scipy.optimize.linprog and sklearn.linear_model.OrthogonalMatchingPursuit, respectively for basis pursuit and orthogonal matching pursuit.[1]

In Figures 5 and 6 we graph, as a function of sparsity $s$ and number of measurements $m$ respectively, the recovery rate to a certain $\ell^2$ norm error threshold, i.e. $\|\mathbf{x}^* - \mathbf{x}\|_2 < 0.01$, where $\mathbf{x}^*$ is the output of the compressed sensing algorithm. Each data point is the fraction of matrices out of the 100 total for which recovery of an $s$-sparse $\mathbf{x}$ was within this error. Note that the set of matrices was the same when we varied $s$, while the $s$-sparse vector $\mathbf{x}$ was re-randomized each time, which corresponds to Figure 5. Then when we varied $m$, the matrices were re-randomized each time, as were the $s$-sparse vectors. In each case we used Gaussian matrices, with $\mathcal{N}(0, \frac{1}{\sqrt{m}})$ entries such that they have $\ell^2$-normalized columns in expectation.

These graphs, though noisy as a result of randomness and low sample size, demonstrate a few key components of compressed sensing, namely that after a certain point recovery is virtually guaranteed, whereas there is a quick drop-off to never being successful. In addition, orthogonal matching pursuit under-performs basis pursuit in both cases, though not to an incredibly large degree. In fact, the utility of orthogonal matching pursuit is that it is usually good enough, while being much faster to run than basis pursuit.

---

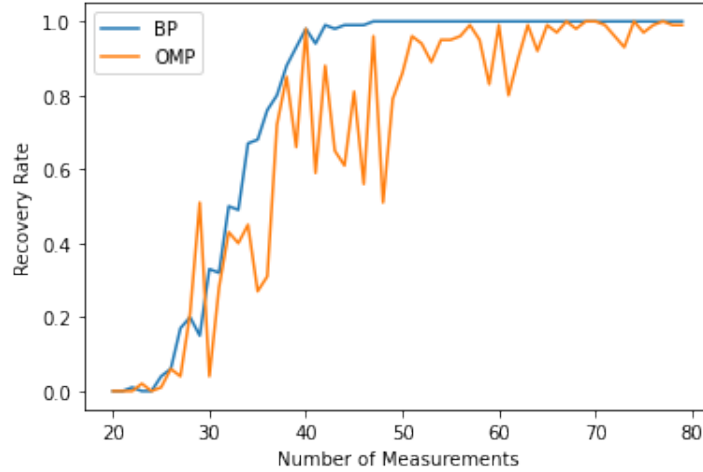[1]Source code available at https://github.com/whartog/senior-thesis-hartog.

FIGURE 6. Recovery rate of random sparse vectors over a set of 100 Gaussian matrices, as a function of number of measurements for fixed $(s = 10, N = 200)$.
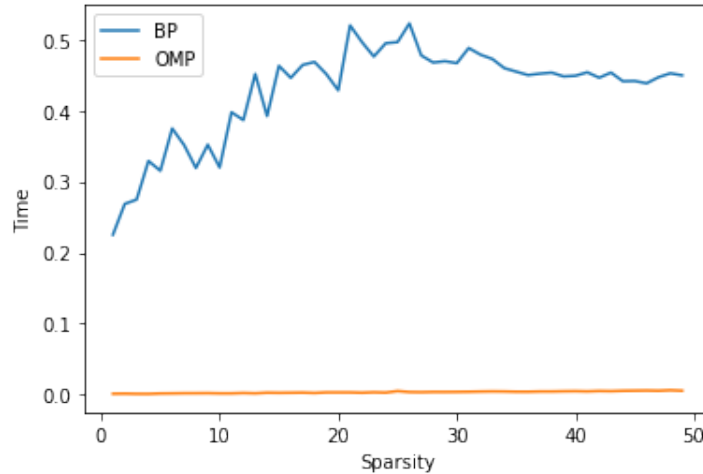


FIGURE 7. The average amount of time it took to run basis pursuit and orthogonal matching pursuit on an individual matrix, for each value of $s$ from Figure 5.

We note also that the higher jaggedness of Figure 6, in comparison to Figure 5, is likely due to the fact that we re-randomized the matrices at every value of $m$ for the former, whereas the matrices for the latter were randomized at the start but constant for every value of $s$. This does however support the notion that recovery success depends on the matrix to a good degree, as the more volatile graph was produced with more randomness in the matrices used.

In Figure 7 we plot the average amount of time, per matrix, that basis pursuit and orthogonal matching pursuit took respectively to run, during the simulation where we varied $s$, which corresponds to Figure 5. Notably, basis pursuit is clearly much more costly in terms of running time as compared with orthogonal matching pursuit. Nevertheless, the low sample size and low values of $N$ and $m$ are significant considerations, as compressed sensing as a method is geared for use on large data sets.
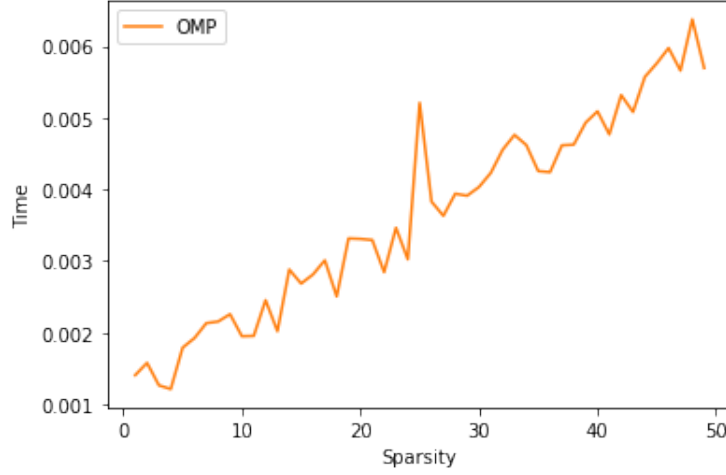
FIGURE 8. The average amount of time it took to run orthogonal matching pursuit on an individual matrix, for each value of $s$ from Figure 5.

In addition, as seen in Figure 8, the main factor for the running time of orthogonal matching pursuit is the sparsity $s$, which makes sense as the algorithm computes $s$ iterations of its core steps. As such, when $s$ is varied, the graph of running time is about linear, meaning that as the data set size and thus $s$ increases, orthogonal matching pursuit will lose some of its advantages over basis pursuit, but the extent to which it was already faster will ensure that it remains the more efficient computation option.

3.5. **A Simulated Comparison of Choices of Measurement Matrix.** We will also use a simulation of our compressed sensing algorithms to obtain a general understanding or at the very least a simple visualization of how the choice of measurement matrix **A** affects the performance of sparse recovery, in particular the recovery rate. In Section 3.4 we saw the performance of basis pursuit using $\ell^2$-normalized Gaussian measurement matrices, so we will use this setting as a baseline against which to compare, which is an especially useful comparison since we have an idea from Theorem 2.9 as to why Gaussian matrices are a good choice. The choice of basis pursuit for the common recovery algorithm also makes sense since it is theoretically the most successful algorithm, though it is slow.

We will simulate two comparisons: the first between Gaussian, Laplace, and Cauchy matrices, and the second between Gaussian and deterministic matrices. First we will discuss the first two of these distributions, and specifically why we include them in our discussion. The Laplace distribution is often known alternatively as the "double exponential" distribution, since its PDF (probability density function) is symmetric about the origin and has the shape of the exponential PDF on either side. As such, the Laplace distribution is an example of a subexponential distribution, which we discussed briefly in Section 2.3. The Cauchy distribution, meanwhile, is defined as the ratio of two independent standard Gaussian random variables. Unlike the Gaussian or Laplace, the Cauchy distribution has a much fatter tail that decays at a rate slower than exponential. The best way to think about this contrast of tails is that the subgaussian and the subexponential have light tails, with the subexponential being one degree heavier, while distributions with polynomial tails have much heavier
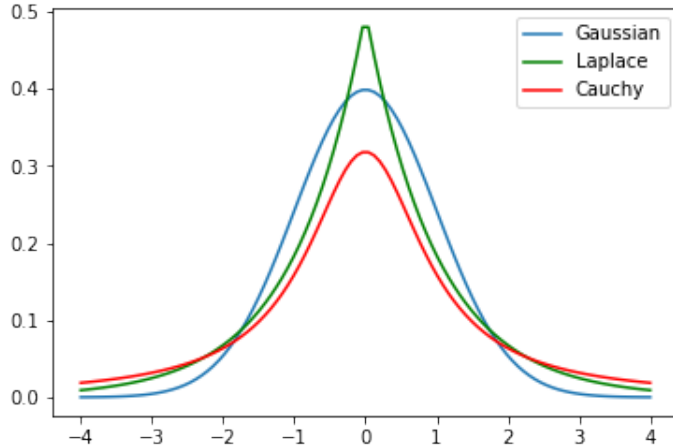
FIGURE 9. Plots of the PDFs for the standard Gaussian, Laplace, and Cauchy distributions. Note that the relative sizes of the distribution tails are determined by how quickly the value of the function decreases as the argument tends towards $\pm\infty$.

tails. Figure 9 has an illustration of the differences between the Gaussian, Laplace, and Cauchy, in terms of the thickness of their tails.

The simulations for Laplace and Cauchy will be identical except for matrix distribution, using the same structure as the simulations from Figure 6, namely of varying the number of measurements $m$, while keeping a fixed $s = 10$ and $N = 200$, and at every value of $m$ computing a new set of 100 random matrices. Both the Laplace and Cauchy matrices will be scaled by a factor of $\frac{1}{\sqrt{m}}$, the same factor by which the Gaussian matrix is scaled. We do this so that we can make a comparison between the performance of the matrices with a clear conscience. If we were to try to scale such that each matrix had $\ell^2$-normalized columns in expectation, then we would have to scale the Laplace matrix by $\frac{1}{\sqrt{2m}}$, since a standard Laplace variable has variance 2, and we would not be able to scale the Cauchy matrix to achieve this end, as the Cauchy distribution does not have a finite mean or variance. As such, we simply use the scaling factor from the Gaussian matrix to maintain some consistency.

Figure 10 shows exactly how thin the margins between subgaussian and supexponential distributions can be. Despite the recovery result in Theorem 2.8, which specifies a subgaussian random measurement matrix setting, Laplace matrices provide an only slightly less successful recovery rate, at least in this isolated experiment, with small values of $s = 10$ and $N = 200$. This graph is an important reminder that the basic theoretical results such as Theorem 2.8 are designed to deal with the worst-case scenario, and are often only necessary when the sample size is much larger than what our experiments use.

However, comparing the recovery rate with Gaussian random matrices versus Cauchy random matrices illustrates why exponentially-decaying tails are a necessity. In Figure 10, we see that while the Cauchy setting eventually reaches a good recovery rate, it requires a much higher number of measurements $m$, which presumably will scale poorly as the dimensionality increases.

We can also visualize via simulation why choosing a random measurement matrix is a much more successful setting than a general deterministic matrix. The following
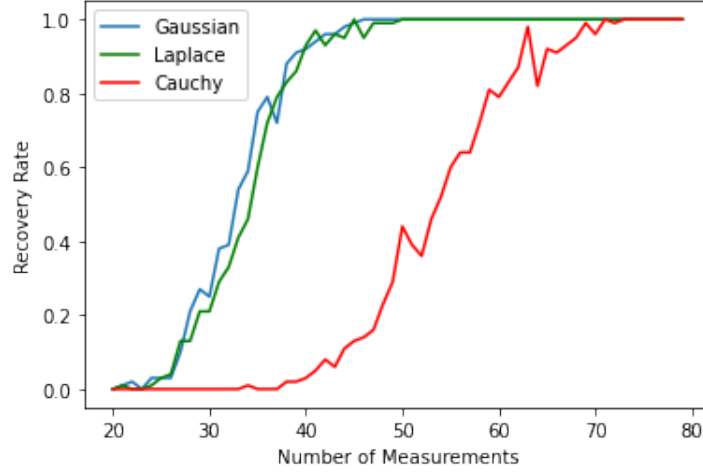
FIGURE 10. Recovery rate of random sparse vectors via basis pursuit for sets of 100 random Gaussian, Laplace, and Cauchy matrices, as a function of number of measurements for fixed ($s = 10, N = 200$).

simulation involves choosing, for each $m$, one random Gaussian $m \times N$ matrix and the deterministic $m \times N$ matrix $\mathbf{A}(m, N)$ chosen according to the following rule:

$$(3.3) \qquad \mathbf{A}_{ij} = i + (j \mod (i + 2)),$$

where $\mathbf{A}_{ij}$ is the entry in the $i$th row and $j$th column of $\mathbf{A}$, with $i \in [m]$ and $j \in [N]$. An example of this matrix, with shape $4 \times 6$, is the following:

$$\mathbf{A}(4, 6) = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 2 & 3 & 1 & 2 & 3 \\ 2 & 3 & 4 & 5 & 2 & 3 \\ 3 & 4 & 5 & 6 & 7 & 3 \end{pmatrix}.$$

This formula for the deterministic matrix was chosen to be a pre-determined formula while producing a matrix without many dependencies between columns or rows, though because $m < N$ the columns will of course be linearly dependent. This choice is in contrast to, say, choosing a matrix of all ones.

Our simulation then takes fixed $s = 10, N = 100$, and produces the plot shown in Figure 11. As for the Laplace and Cauchy matrices, we scale both the Gaussian and the deterministic matrix by a factor of $\frac{1}{\sqrt{m}}$ just so the comparison is valid. While the Gaussian setting experiences total success soon after $m = 30$, the deterministic matrix is unable to exceed a success rate of 0.5. While we might expect this success rate to increase as $m$ continues to increase, it is clear that this particular choice of determinstic matrix is a very poor choice of measurement matrix, and will likely become much worse as the dimensionality of the problem increases.

After only investigating this single particular choice of deterministic matrix, we do not have enough evidence to generalize to the assertion that any deterministic matrix will fail, which is in fact not the case. In fact, the matrices we are technically deterministic, as they are only *pseudorandom*, as computers do not have the capacity to produce truly random values. In addition, other formulations for deterministic measurement matrices which work well for sparse recovery are possible, including
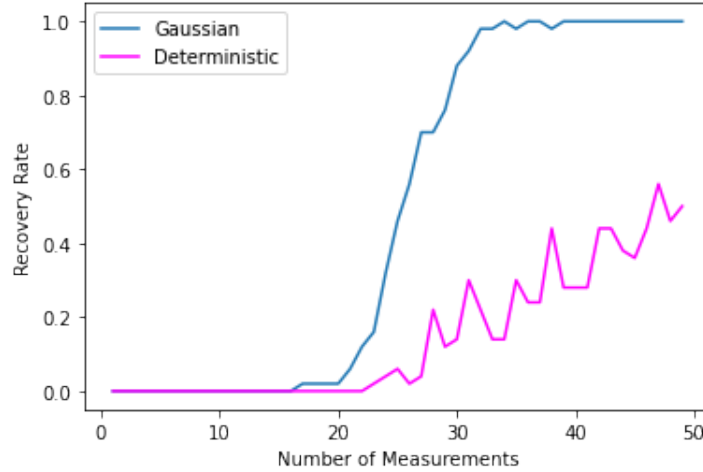
FIGURE 11. Recovery rate via basis pursuit of a set of 50 $s$-sparse vectors, for a single random Guassian or deterministic matrix defined according to (3.3), as a function of number of measurements for ($s = 10, N = 100$).

Fourier frames [8], which can be empirically shown to perform well for sparse recovery. However, the main barrier in this regard, which is a crucial part of ongoing research, is how to prove that such deterministic settings work [8]. In any case, it is clear from Figure 11 that it is certainly possible to produce deterministic measurement matrices which are poorly suited for compressed sensing, despite efforts to construct them in a somewhat pseudorandom-looking way.

## 4. Compressed Sensing in Application: MRI Technology

So far we have detailed the initial theory of compressed sensing, involving the conditions under which exact recovery succeeds and the algorithms which are often used to accomplish this task. However, the main utility of compressed sensing lies in its application, and there are many more complications to keep in mind when working with real-world data. Two particular considerations are stable and robust recovery, the former being a notion of success even when the vectors are not exactly sparse, and the latter being a notion of success even when taking into account measurement error [8].

As such, we aim in this section to detail a specific application of compressed sensing in order to better understand the challenges of applying the method. The application area in question is magnetic resonance imaging (MRI), which is most well known for its use in medical situations, including for imaging the brain or spine. The technology makes use of an external magnetic field to get certain protons to emit radio frequencies which can then be tracked and coalesced into images. In particular, these protons often come from water or fat molecules in the human body. For a more in-depth understanding of the physics behind MRI the interested reader may refer to [19].

For our purposes the key takeaway is that traditionally MRI machines needed to take many samples in order to fully form the desired images, which was costly, both in terms of time spent and the physical cost of the equipment. Thus, the work of Lustig et al. in 2007, in the aftermath of the seminal compressed sensing literature, made the crucial step of taking advantage of the inherent sparse structure of MRI images

to apply compressed sensing methods [12]. We discuss the general mathematical definition of the MRI problem in Section 4.1 and overview the way in which one can harness the sparse structure of MRI in Section 4.2. As of the time of writing, this advancement in MRI technology remains one of the most significant successes of compressed sensing and a signature example of its utility [19].

4.1. **Mathematical Formulation of MRI.** The subsequent notation we introduce for interpreting the MRI image acquisition problem is adapted from Wright and Ma [19].

While the exact means by which a "signal" is produced relies on the physics (and for a more detailed exposition in this regard we refer the reader to [19]), the ultimate product is a function in two variables $\mathbf{M}(x, y)$ which at a high level represents the magnetic field at coordinates $(x, y)$ for a specific third axis value $z_0$. In order to obtain a complete picture of the target, we produce a signal computed with the magnitude $|\mathbf{M}(x, y)|$ defined at time $t$ by

$$(4.1) \qquad s(t) = e^{-i\omega_0 t} \int_x \int_y |\mathbf{M}(x, y)| e^{-i(k_x(t)x + k_y(t)y)} dx dy,$$

where $\omega_0$ is a set frequency known in the literature as the *Larmor frequency*, and the functions $k_x, k_y$ form a transformation from $(x, y)$ to what is known as *k-space*, based on the properties of the gradient of a second magnetic field [19]. Then re-expressing this function with regard to values $k_x, k_y$ and ignoring the $e^{-\omega_0 t}$ term we get

$$(4.2) \qquad S(k_x, k_y) = \int_x \int_y |\mathbf{M}(x, y)| e^{-i(k_x x + k_y y)} dx dy,$$

which is essentially the 2D Fourier transform of $|\mathbf{M}(x, y)|$ at $k_x, k_y$. For some intuition regarding the presence of a Fourier transform, note that the purpose of a Fourier transform is often to move between the spatial and frequency domain, and the MRI measurement process is better equipped to work in the frequency domain. In addition, note that this formulation motivates our insistence on giving compressed sensing results in complex dimensions (i.e. $\mathbf{x} \in \mathbb{C}^N, \mathbf{y} \in \mathbb{C}^m, \mathbf{A} \in \mathbb{C}^{m \times N}$), since the function of complex Fourier transform values as measurements is a natural setting.

With this Fourier structure, it is then a natural goal to want to use an inverse Fourier transform to recover the image $I(x, y)$ (in the form of the function $|M(x, y)|$), with enough measurements at various frequencies $\{(k_x(t_i), k_y(t_i)\}_{i \in I}$ for some index set $I$ corresponding to different times $\{t_i\}_{i \in I}$.

Recovering a continuous function however is difficult and unnecessary, since any image we produce will have some finite resolution. Thus, we treat $I(x, y)$ as a function on an $N \times N$ grid, and call $\mathbf{u} = (k_x, k_y)$ the vector of coordinates across the chosen times [19], in which case our measurement will be of the form

$$(4.3) \qquad \mathbf{y}(\mathbf{u}) = \mathcal{F}[I(x, y)](\mathbf{u}),$$

where $\mathcal{F}$ is the discrete Fourier transform, in the form of an $m \times m$ matrix $\mathcal{F}[I(x, y)]$ where $m = N^2$. Then the image at point $(x, y)$ is recovered via the inverse discrete Fourier transform at $\mathbf{y}(\mathbf{u})$.

In is however redundant to recover these $I(x, y)$ values, of which there are $m$, so sampling the entire transform domain is not necessary. Meanwhile, traditional signal processing knowledge, from our discussion of Theorem 0.1 (the Nyquist sampling rate), says that if our frequencies are bounded above by $f_{\max}$, then we only need to sample at a rate of over $2f_{\max}$ frequencies. Yet this rate can still be costly, as we

discussed in the introduction, so based on the initial work by Lustig et al. [12], we look for a notion of sparsity with the goal of applying compressed sensing techniques.

4.2. **Translating MRI to a Compressed Sensing Setting.** The "sparsity" of MR images is a illuminating example of the power of sparse thinking, in that these images are certainly not sparse, given most of their pixels are nonzero, but they do have a well-defined structure, which leads to sparsity in a given basis. In particular, MR images (and many natural images), when viewed as functions in 2D space, are sparse in the wavelet transform domain [12], which is associated with a collection $\{\psi_i\}_{i \in [N^2]}$ of functions on 2D space [19]. We reserve a full explanation of the definition of wavelets and why images are sparse in the wavelet domain to the literature [12], but an intuitive picture to keep in mind is that most images of a certain type follow various patterns that allow them to be represented by a combination of smaller set pieces.

To formalize the function of this wavelet basis, we think of $\Phi$ taking an image to a representation $\mathbf{x}$ in the wavelet domain: $\Phi[I] = \mathbf{x}$, with the natural inverse $\Psi = \Phi^{-1}$ sending the vector $\mathbf{x}$ of coefficients to a 2D image:

$$I = \Psi[\mathbf{x}] = \sum_{i \in [N^2]} \mathbf{x}_i \psi_i.$$

The notion of sparsity for this representation is that a small number of the coefficients $\mathbf{x}_i$ are nonzero. There are other alternatives to the wavelet basis [19], but it is one of the clearer and more often used options, and as such we proceed to give a compressed sensing formulation which takes advantage of this newly found sparse representation.

Using this representation of $I$, we can express our measurement $\mathbf{y}$ in (4.3) as

$$(4.4) \qquad\qquad\qquad \mathbf{y} = \mathcal{F}[\Psi \mathbf{x}],$$

where $\mathbf{z} := \Psi \mathbf{x}$ is a vector with shape $N^2$ representing $I$. Taking $m \ll N^2$ measurements we can produce a $m \times N^2$ Fourier matrix $\mathcal{F}_m \in \mathbb{C}^{m \times N^2}$, so we can re-formulate (4.4) as

$$\mathbf{y} = \mathcal{F}_m \Psi \mathbf{x} = \mathbf{A} \mathbf{x}$$

where $\mathbf{A} := \mathcal{F}_m \Psi$ is now a measurement matrix which completes the set of measurements necessary for compressed sensing. It turns out that when the $m$ measurements are sampled randomly from the full $N^2 \times N^2$ Fourier matrix $\mathcal{F}$, the matrix $\mathbf{A}$ is sufficiently incoherent and then by our coherence-related recovery results in Section 2.1 our compressed sensing methods will work. However, due to the distribution of the wavelet coefficients a density-based sampling scheme, as opposed to a uniformly random one, is better to use [19]. Once a vector $\mathbf{x}^*$ is recovered via compressed sensing, the image is recovered by applying $\Psi : \mathbf{z}^* = \Psi \mathbf{x}^*$.

As with any application area, feasibility concerns will arise and necessitate alterations to the ideal compressed sensing setting. A primary challenge for MRI involves the infeasibility of a fully random sampling scheme, namely that an MRI machine cannot sample at completely random locations, instead able to sample along trajectories in the $k$-space [19]. Thus a few common solutions include to sample a set of radii or a set of concentric circles from the center of the image.

A further concern is measurement error, which is common in the MRI setting in the form of thermal noise, a natural outcome of the high-power technology utilized

for the method. The subsequent minimization problem accounts for this possibility by bounding the $\ell^2$ measurement error by $\eta$:

$$\text{minimize } \|\mathbf{z}\|_1 \text{ subject to } \|\mathbf{Az} - \mathbf{y}\|_2 \leq \eta,$$

which is equivalent to a notion of *robust* recovery [19, 8]. Robust recovery, while not discussed to a great extent in this paper, is a naturally important consideration for any application of compressed sensing due to the ubiquity of noise in real-world measurements.

## 5. Conclusion and Next Steps for Compressed Sensing

In this paper we have given a review of the early theory for compressed sensing, including the intuition for why the method works, the conditions under which sparse recovery is successful, and the main algorithms used in its implementation. We have also introduced an example of the application of compressed sensing to MRI technology.

However, the theory presented is almost entirely preoccupied with the most ideal case, being of exact recovery for exactly sparse signals, two assumptions which are both unrealistic. These issues obtain consideration in the form of robust and stable recovery respectively, which were mentioned briefly in Section 4. These considerations are important since the main work currently going into compressed sensing is in formatting the theory for application. Now that the significance of sparsity and structure are well understood and the dust has settled on the initial flurry of seminal work during the mid to late 2000's, compressed sensing has its place as a pillar of practical signal processing.

Alongside MRI technology, which was one of the earliest applications of compressed sensing, and might be considered the most significant, the field has been applied to other areas of imaging, such as radar [20]. Additionally, a signature new technology born out of compressed sensing is the so-called single-pixel camera, which was built to take advantage of sparsity assumptions in order to make simpler measurements and as a result be much more efficient [7]. Compressed sensing is also extensively used in the field of communications, in many different ways, making the field one of the more robust areas of application for compressed sensing [3].

Due to the ubiquity of its core assumption of sparsity and in particular, structure, which is found in most natural phenomena, researchers have proposed compressed solutions to many additional problems, including facial recognition, video processing, and speech processing. Of course each new set of data provides its own challenge, a flavor for which we got through our discussion of MRI technology. However, there are a few common challenges that remain significant frontiers for the field.

Perhaps the most significant frontier to conquer is a general acquisition scheme for measurements [15]. As mentioned in Section 2.3, the most common setting for compressed sensing is with a random measurement matrix. Constructing deterministic matrices for which one can prove their optimality in the sparse recovery sense is still a difficult open problem [8]. Since true randomness is hard to accomplish, especially given the limitations of many real-world technologies such as MRI machines, formulating a measurement scheme which achieves the requisite measurement matrix conditions, often through a pseudorandom process, is of the utmost importance. As per mathematician Robert Coveyou's famous titular adage: "random number generation is too important to be left to chance" [5].

One potential alternative that has been in the works for about a decade is structured compressed sensing, which approaches the issue of constructing a suitable measurement matrix by taking a so-called "structured" matrix, which is produced by taking the matrix to be a basis which is orthonormal to the basis in which the signal is sparse [8]. Among the many areas in which structured compressed sensing can be utilized is in image resolution. However, as with any method there are trade-offs, with structured compressed sensing being an overall simpler method with a faster measurement process, while needing more measurements and having a harder time satisfying the restricted isometry property [15]. As such some researchers are looking for ways to provide for sparse recovery while bypassing the restricted isometry property, which is usually the most common theoretical recovery guarantee.

As we saw with MRI, there are also many application-specific challenges that arise, including a necessity for new algorithms, a universal measurement process, and more efficient computation. These challenges just serve to keep compressed sensing, though certainly past its heyday of the late 2000's and early 2010's, an intriguing topic for mathematicians and engineers alike, with its rigorous mathematical theory and rich application power. In the modern world of big data, where efficient computation is at a premium, it is a field that we cannot afford to overlook.

## A. Recovery Conditions for Basis Pursuit and Orthogonal Matching Pursuit

In the first part we will develop the theory necessary to prove Theorems 2.4 and 2.2, with respect to the success of basis pursuit, which as previously mentioned is the $\ell^1$-minimization (0.3). This theory is adapted from Foucart and Rauhut [8].

Before we introduce Theorem A.2 which will allow us to complete these proofs in Section 2, we introduce a third notion of matrix niceness, alongside incoherence and the restricted isometry property, known as the null space property. Note that we present the null space property here and not in the main body since it is most useful as an intermediate step, whereas coherence and the restricted isometry constant are easier to work with.

**Definition A.1.** A matrix $\mathbf{A} \in \mathbb{C}^{m \times N}$ satisfies the *null space property* relative to a set $S \subset [N]$ if

$$(A.1) \qquad \|\mathbf{v}_S\|_1 < \|\mathbf{v}_{\bar{S}}\|_1 \text{ for all nonzero } \mathbf{v} \in \ker(\mathbf{A}).$$

We obtain a subsequent reformulation of this property (A.1) by adding $\|\mathbf{v}_S\|_1$ to both sides of the inequality.

$$(A.2) \qquad 2\|\mathbf{v}_S\|_1 < \|\mathbf{v}\|_1 \text{ for all nonzero } \mathbf{v} \in \ker(\mathbf{A}).$$

We can also define the null space property *of order $s$*, which holds for a matrix $\mathbf{A}$ if $\mathbf{A}$ satisfies the null space property relative to $S$ for all card$(S) \leq s$. Then the natural connection of the null space property to $\ell^1$-minimization is given by the following theorem, which we also prove.

**Theorem A.2.** *For a matrix $\mathbf{A} \in \mathbb{C}^{m \times N}$, every $s$-sparse $\mathbf{x} \in \mathbb{C}^N$ is uniquely recovered by basis pursuit with $\mathbf{y} = \mathbf{Ax}$ if and only if $\mathbf{A}$ satisfies the null space property of order $s$.*

*Proof.* First we assume that basis pursuit uniquely recovers $\mathbf{x}$ with the given inputs, so any $s$-sparse $\mathbf{x}$ is the unique minimizer of $\|\mathbf{z}\|_1$ under the constraint that $\mathbf{Az} = \mathbf{Ax}$. Thus, for any nonzero vector $\mathbf{v} \in \ker(\mathbf{A})$, we can take any index set $S \subset [N]$ with $\mathrm{card}(S) \leq s$ and $\mathbf{v}_S$ will be an $s$-sparse vector meaning it is the unique minimizer of $\|\mathbf{z}\|_1$ under the constraint that $\mathbf{Az} = \mathbf{Av}_S$. However, $\mathbf{A}(\mathbf{v}_S + \mathbf{v}_{\bar{S}}) = \mathbf{Av} = \mathbf{0}$ and thus $\mathbf{A}(-\mathbf{v}_{\bar{S}}) = \mathbf{Av}_S$ where $-\mathbf{v}_{\bar{S}} \neq \mathbf{v}_S$ as $\mathbf{v} \neq \mathbf{0}$, so $\|\mathbf{v}_S\|_1 < \|\mathbf{v}_{\bar{S}}\|_1$ since $\mathbf{v}_S$ is the unique minimizer.

Next we assume that $\mathbf{A}$ satisfies the null space property of order $s$, so for any nonzero $\mathbf{v} \in \ker(\mathbf{A})$ and index set $S \subset [N]$ with $\mathrm{card}(S) \leq s$, $\|\mathbf{v}_S\|_1 < \|\mathbf{v}_{\bar{S}}\|_1$. Then taking $\mathbf{x}$ $s$-sparse, we will show that $\mathbf{x}$ is the unique minimizer of $\|\mathbf{z}\|_1$ with the constraint that $\mathbf{Az} = \mathbf{Ax}$. For $\mathbf{z} \neq \mathbf{x}$ satisfying $\mathbf{Az} = \mathbf{Ax}$, we can define the nonzero vector $\mathbf{v} := \mathbf{x} - \mathbf{z} \in \ker(\mathbf{A})$, which allows us to bound:

$$\|\mathbf{x}\|_1 \leq \|\mathbf{v}_S\|_1 + \|\mathbf{z}_S\|_1 < \|\mathbf{v}_{\bar{S}}\|_1 + \|\mathbf{z}_S\|_1 = \|-\mathbf{z}_{\bar{S}}\|_1 + \|\mathbf{z}_S\|_1 = \|\mathbf{z}\|_1.$$

This utilizes the fact that $\mathbf{x}$ is only supported on $S$ for the first inequality, and for the equality that $\mathbf{v}_{\bar{S}} = -\mathbf{z}_{\bar{S}}$. $\qquad\square$

At this point we are done, with what was a fairly clean proof which provides us the tools necessary to prove the basis pursuit parts of Theorems 2.4 and 2.2.

Next we will state and prove the proposition that allows us to prove the orthogonal matching pursuit portion of Theorem 2.2, the statement and proof of which are likewise adapted from Foucart and Rauhut [8].

**Proposition A.3.** *For a matrix $\mathbf{A} \in \mathbb{C}^{m \times N}$, each nonzero $\mathbf{x} \in \mathbb{C}^N$ supported on an index set $S \subset [N]$ with $\mathrm{card}(S) = s$ is recovered from input measurement vector $\mathbf{y} = \mathbf{Ax}$ after at most $s$ iterations of orthogonal matching pursuit if and only if $\mathbf{A}_S$ is injective and for all nonzero $\mathbf{w} = \mathbf{Az}$ with $\mathrm{supp}(\mathbf{z}) \subset S$,*

$$(A.3) \qquad \max_{j \in S} |(\mathbf{A}^*\mathbf{w})_j| > \max_{k \notin S} |(\mathbf{A}^*\mathbf{w})_k|.$$

Here $\mathbf{A}_S$ is the matrix $\mathbf{A}$ constrained to only the columns corresponding to the $s$ indices in the set $S$, and $\mathbf{a}_i$ is the $i$th column of the matrix $\mathbf{A}$. Note that $(\mathbf{A}^*\mathbf{w})_i = \langle \mathbf{w}, \mathbf{a}_i \rangle$, the latter of which we used in our proof of Theorem 2.2. Next we will prove the above proposition.

*Proof.* Assuming first that orthogonal matching pursuit recovers each nonzero $\mathbf{x}$ in the given way, we see that the matrix $\mathbf{A}_S$ is injective since any vectors supported on $S$ which have the same measurement vector must be the same (otherwise orthogonal matching pursuit might not output the correct vector). Then we also know that the first index chosen by orthogonal matching pusuit is in the target index set $S$, meaning $\max_{j \in S} |(\mathbf{A}^*\mathbf{y})_j| > \max_{k \notin S} |(\mathbf{A}^*\mathbf{y})_k|$, which will be true for any $\mathbf{y} = \mathbf{Ax}$ for $\mathbf{x}$ supported on $S$, so we are done with the forwards direction.

Next we assume that $\mathbf{A}_S$ is injective and (A.3) holds for all relevant $\mathbf{w}$. We need to show that in the context of the orthogonal matching pursuit notation given in Section 3.2, $S^n \subset S$ for all $n \in [s]$ and thus $S^s = S$, at which point $\mathbf{x}^s = \mathbf{x}$ by the injectivity of $\mathbf{A}_S$, correctly recovering the unique $s$-sparse signal $\mathbf{x}$. The base case is $S^0 = \varnothing \subset S$, so we proceed inductively, assuming $S^n \subset S$ for some $0 \leq n \leq s-1$, meaning $\mathbf{w}^n := \mathbf{y} - \mathbf{Ax}^n = \mathbf{Az}$ for some $s$-sparse $\mathbf{z}$. Thus by (A.3) the next index $j_{n+1} \in S$ meaning $S^{n+1} \subset S$, so we are done. $\qquad\square$

## B. Inequalities Involving the Restricted Isometry Constant and Coherence

First we give an alternative definition of the restricted isometry constant, using the matrix norm [8].

**Definition B.1.** The $s$th restricted isometry constant $\delta_s$ of a matrix $\mathbf{A} \in \mathbb{C}^{m \times N}$ is

$$\delta_s := \max_{S \subset [N], \text{card}(S) \leq s} \|\mathbf{A}_S^* \mathbf{A}_S - \mathbf{Id}\|_{2 \to 2}.$$

Here the $*$ operator is the conjugate transpose, and the $\| \cdot \|_{2 \to 2}$ norm is the operator norm from $\ell^2$ to $\ell^2$ defined as the maximum singular value of that matrix. The measure here intuitively gives the worst case scenario, under index sets $S$ with $\text{card}(S) \leq s$, of how close $\mathbf{A}_S$, the restriction of $\mathbf{A}$ to the index set $S$, is to an isometry. With this definition we may then prove the following proposition relating the restricted isometry constant to coherence.

**Proposition B.2.** *If a matrix $\mathbf{A} \in \mathbb{C}^{m \times N}$ has $\ell^2$-normalized columns, then its coherence and restricted isometry constants satisfy*

$$\delta_1 = 0, \quad \delta_2 = \mu, \quad \delta_s \leq \mu_1(s-1) = (s-1)\mu, \quad s \geq 2.$$

To do this first we must understand the meaning of $\mu$, which is a more general notion of coherence, not associated with a sparsity $s$: $\mu := \max_{1 \leq i \neq j \leq N} |\langle \mathbf{a}_i, \mathbf{a}_j \rangle|$ is the coherence of $A$, where $\mathbf{a}_i$ are the columns of $\mathbf{A}$. Now we will prove the first two parts Proposition B.2, and cite a result for the third.

*Proof.* First, the only index sets of size 1 are $S = \{j\}$, so for any such $S$, $\mathbf{A}_S^* \mathbf{A}_S = \mathbf{a}_j^* \mathbf{a}_j = \|\mathbf{a}_j\|_2 = 1$, since the columns $\mathbf{a}_j$ are normalized. The 1-dimensional identity matrix is just 1, so $\delta_1 = 0$.

For $\delta_2$, we need to compute $\mathbf{A}_{\{i,j\}}^* \mathbf{A}_{\{i,j\}}$ for any $i \neq j$ with $\{i,j\} \subset [N]$, which will be

$$\mathbf{A}_{\{i,j\}}^* \mathbf{A}_{\{i,j\}} = \begin{pmatrix} 1 & \langle \mathbf{a}_j, \mathbf{a}_i \rangle \\ \langle \mathbf{a}_i, \mathbf{a}_j \rangle & 1 \end{pmatrix} \Rightarrow \mathbf{A}_{\{i,j\}}^* \mathbf{A}_{\{i,j\}} - \mathbf{Id} = \begin{pmatrix} 0 & \langle \mathbf{a}_j, \mathbf{a}_i \rangle \\ \langle \mathbf{a}_i, \mathbf{a}_j \rangle & 0 \end{pmatrix}.$$

The eigenvalues of this matrix are $\pm |\langle \mathbf{a}_i, \mathbf{a}_j \rangle|$, so $\|\mathbf{A}_{\{i,j\}}^* \mathbf{A}_{\{i,j\}} - \mathbf{Id}\|_{2 \to 2} = |\langle \mathbf{a}_i, \mathbf{a}_j \rangle|$ and thus

$$\delta_2 = \max_{i \neq j} |\langle \mathbf{a}_i, \mathbf{a}_j \rangle| = \mu$$

as already defined. The $s \geq 2$ case can be shown using Theorem 5.3 from [8], though I have the space to include neither it nor the proof thereof. $\square$

Proposition B.2, along the fact that the restricted isometry constant is non-decreasing in $s$, gives us the result (2.4), since substituting $s+1$ for $s$ (where $s \geq 1$), we have that $\delta_{s+1} \leq \mu_1(s) = s\mu$, where $\mu = \delta_2 \leq \delta_{s+1}$.

## C. Basis Pursuit via Linear Programming

Linear programming is a fairly standard part of optimization literature, having a dedicated function (scipy.optimize.linprog) within the SciPy library in Python, which is what we used to implement basis pursuit in Section 3.4. A classic linear programming formulation, which is the one employed in the SciPy documentation is as follows:

(C.1)  $\qquad \text{argmin}_{\mathbf{x}} \, \mathbf{c}^\top \mathbf{x}$  such that $\mathbf{A}_{ub}\mathbf{x} \leq b_{ub}, \mathbf{A}_{eq}\mathbf{x} = \mathbf{b}_{eq}, \mathbf{l} \leq \mathbf{x} \leq \mathbf{u},$

where $\mathbf{c}, \mathbf{l}, \mathbf{u}$ are all vectors of the same shape as $\mathbf{x}$, $\mathbf{A}_{ub}, \mathbf{A}_{eq}$ are matrices, possibly of different shapes, and $\mathbf{b}_{ub}, \mathbf{b}_{eq}$ are vectors of corresponding shapes to work with the matrix multiplication. An interpretation of the problem is to minimize some weighted version of $\mathbf{x}$ (which is distinct from the interpretation from the compressed sensing literature), subject to various linear constraints regarding $\mathbf{x}$, which form a feasible set in the form of a convex polytope. The standard formulation often omits the equality constraint, since it can be achieved through a re-formulation of the inequality [4], but the SciPy formulation is easier to use. Solving this problem (C.1) is actually relatively straightforward to code, a more detailed explanation of which can be found in [4].

In this appendix we restrict ourselves to simply showing how to formulate $\ell^1$-minimization (0.3), or basis pursuit, using linear programming. In particular, because basis pursuit minimizes $\|\mathbf{z}\|_1$, which is a sum of absolute entries, we need to produce a new formulation that does not use any absolute values. Thus, in addition to $\mathbf{z}$ we consider the vector $\mathbf{t}$, which we interpret as the absolute upper bound of $\mathbf{z}$, i.e. $|\mathbf{z}_i| \leq \mathbf{t}_i$ for all $i \in [N]$. Thus, what we actually want to minimize is $\sum_i \mathbf{t}_i$, subject to $\mathbf{A}\mathbf{z} = \mathbf{y} = \mathbf{A}\mathbf{x}$ as the equality constraint, and with $\mathbf{z}_i \leq \mathbf{t}_i, \mathbf{z} \geq -\mathbf{t}_i$ for all $i$, or in other words $I_N\mathbf{z} - I_N\mathbf{t} \leq \mathbf{0}$ and $-I_N\mathbf{z} - I_N\mathbf{t} \leq \mathbf{0}$.

To use all of these constraints we need to combine $\mathbf{z}$ and $\mathbf{t}$ into one vector $\mathbf{w} = (\mathbf{z}_1, ..., \mathbf{z}_N, \mathbf{t}_1, ..., \mathbf{t}_N)^\top$. Thus, our linear program takes the following inputs:

$$\mathbf{c} = (0, ..., 0, 1, ..., 1)^\top, \mathbf{A}_{eq} = \begin{pmatrix} \mathbf{A} \\ \mathbf{0} \end{pmatrix}, \mathbf{b}_{eq} = \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix}, \mathbf{A}_{ub} = \begin{pmatrix} I_N & -I_N \\ -I_N & -I_N \end{pmatrix}, \mathbf{b}_{ub} = \mathbf{0}.$$

They have respective shape $2N \times 1$ for the $\mathbf{c}$ of $N$ zeroes and $N$ ones, $2m \times N$ for $\mathbf{A}_{eq}$, $2m \times 1$ for $\mathbf{b}_{eq}$, $2N \times 2N$ for $\mathbf{A}_{ub}$, and $2N \times 1$ for $\mathbf{b}_{ub}$. No $\mathbf{l}, \mathbf{u}$ values are required as the constraints of basis pursuit were already implement by the previous constraints.

## References

[1] Agresti, A. (2015). Foundations of Linear and Generalized Linear Models. Hoboken, NJ: Wiley.

[2] Candès, E.J., Romberg, J., Tao, T. (2006). Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. IEEE Transactions on Information Theory, 52(2), 489-509.

[3] Choi, J.W., Shim, B., Ding, Y., Rao, B., Kim, D.I. (2017). IEEE Communications Surveys Tutorials, 19(3), 1527-1550.

[4] Chvátal, V. (1983). Linear Programming. New York, NY: W.H. Freeman and Company.

[5] Coveyou, R.R. (1969). Random number generation is too important to be left to chance. Studies in Applied Mathematics, 3, 70-111.

[6] Donoho, D.L. (2006). Compressed sensing. IEEE Transactions on Information Theory, 52(4), 1289-1306.

[7] Duarte, M.F., Davenport, M.A., Takhar, D., Laska, J.N., Sun, T., Kelly, K.F., Baraniuk, R.G. (2008). IEEE Signal Processing Magazine, 25(2), 83-91.

[8] Foucart, S., Rauhut, H. (2013). A Mathematical Introduction to Compressive Sensing. New York, NY: Birkhäuser.

[9] Hazimeh, H., Mazumder, R., Saab, A. (2020). Sparse Regression at Scale: Branch-and-Bound rooted in First-Order Optimization. arXiv pre-print. http://arxiv.org/abs/2004.06152.

[10] Kunis, S., Rauhut, H. (2008). Random Sampling of Sparse Trigonometric Polynomials, II. Orthogonal Matching Pursuit versus Basis Pursuit. Foundations of Computational Mathematics, 8(6), 737-763.

[11] Logan, B. (1965). Properties of high-pass signals. P.h.D. dissertation, Columbia University.

[12] Lustig, M., Donoho, D., Pauly, J.M. (2007). Sparse MRI: The Application of Compressed Sensing for Rapid MR Imaging. Magnetic Resonance in Medicine, 58(6), 1182-1195.

[13] Narayanan, S., Sahoo, S. K., Makur, A. (2015). Greedy Pursuits Assisted Basis Bursuit for Compressive Sensing. 2015 23rd European Signal Processing Conference, 694-698.

[14] Park, S., Lee, H. (2011). On the Derivation of RIP for Fandom Gaussian Matrices and Binary Sparse Signals. 2011 International Conference on ICT Convergence, 120-124.

[15] Rani, M., Dhok, S.B., Deshmukh, R.B. (2018). A Systematic Review of Compressive Sensing: Concepts, Implementations and Applications. IEEE Access, 6, 4875-4894.

[16] Tropp, J. A., Gilbert, A. C. (2007). Signal Recovery From Random Measurements Via Orthogonal Matching Pursuit. IEEE Transactions on Information Theory, 53(12), 4655-4666.

[17] Vershynin, R. (2015). "Estimation in High Dimensions: A Geometric Perspective", Sampling Theory, a Renaissance: Compressive Sensing and Other Developments. Springer International Publishing, 3-66.

[18] Wang, X. (2005). Volumes of Generalized Unit Balls. Mathematics Magazine, 78(5), 390-395.

[19] Wright, J., Ma, Y. (2021). High-Dimensional Data Analysis with Low-Dimensional Models: Principles, Computation, and Applications. Cambridge University Press.

[20] Yang, J., Jin, T., Xiao, C., Huang, X. (2019). Sensors, 19(14), 3100.

HARVARD UNIVERSITY, DEPARTMENTS OF MATHEMATICS AND STATISTICS, CAMBRIDGE MA 02138

*Email address*: `whartog@college.harvard.edu`