Applied and Numerical Harmonic Analysis

Holger Boche
Giuseppe Caire
Robert Calderbank
Gitta Kutyniok
Rudolf Mathar
Philipp Petersen
Editors

Compressed Sensing and Its Applications

Third International MATHEON Conference 2017

Birkhäuser

Birkhäuser

Applied and Numerical Harmonic Analysis

More information about this series at http://www.springer.com/series/4968

Holger Boche · Giuseppe Caire ·
Robert Calderbank · Gitta Kutyniok ·
Rudolf Mathar · Philipp Petersen
Editors

Compressed Sensing and Its Applications

Third International MATHEON Conference 2017

Birkhäuser

Editors

Holger Boche
Department of Electrical and Computer Engineering, Munich Center for Quantum Science and Technology (MCQST)
Technical University of Munich
Munich, Germany

Robert Calderbank
Department of Electrical and Computer Engineering
Duke University
Durham, NC, USA

Rudolf Mathar
Faculty of Electrical Engineering and Information Technology
RWTH Aachen University
Aachen, Nordrhein-Westfalen, Germany

Giuseppe Caire
Institute of Telecommunications Systems
Technical University of Berlin
Berlin, Germany

Gitta Kutyniok
Department of Mathematics
Technical University of Berlin
Berlin, Germany

Philipp Petersen
Mathematical Institute
University of Oxford
Oxford, UK

ISSN 2296-5009 ISSN 2296-5017 (electronic)
Applied and Numerical Harmonic Analysis
ISBN 978-3-319-73073-8 ISBN 978-3-319-73074-5 (eBook)
https://doi.org/10.1007/978-3-319-73074-5

Mathematics Subject Classification (2010): 68P30, 94A08, 94A12, 94A20, 90C90

This book is published under the imprint Birkhäuser, www.birkhauser-science.com by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

ANHA Series Preface

The *Applied and Numerical Harmonic Analysis (ANHA)* book series aims to provide the engineering, mathematical, and scientific communities with significant developments in harmonic analysis, ranging from abstract harmonic analysis to basic applications. The title of the series reflects the importance of applications and numerical implementation, but richness and relevance of applications and implementation depend fundamentally on the structure and depth of theoretical underpinnings. Thus, from our point of view, the interleaving of theory and applications and their creative symbiotic evolution is axiomatic.

Harmonic analysis is a wellspring of ideas and applicability that has flourished, developed, and deepened over time within many disciplines and by means of creative cross-fertilization with diverse areas. The intricate and fundamental relationship between harmonic analysis and fields such as signal processing, partial differential equations (PDEs), and image processing is reflected in our state-of-the-art *ANHA* series.

Our vision of modern harmonic analysis includes mathematical areas such as wavelet theory, Banach algebras, classical Fourier analysis, time-frequency analysis, and fractal geometry, as well as the diverse topics that impinge on them.

For example, wavelet theory can be considered an appropriate tool to deal with some basic problems in digital signal processing, speech and image processing, geophysics, pattern recognition, biomedical engineering, and turbulence. These areas implement the latest technology from sampling methods on surfaces to fast algorithms and computer vision methods. The underlying mathematics of wavelet theory depends not only on classical Fourier analysis, but also on ideas from abstract harmonic analysis, including von Neumann algebras and the affine group. This leads to a study of the Heisenberg group and its relationship to Gabor systems, and of the metaplectic group for a meaningful interaction of signal decomposition methods. The unifying influence of wavelet theory in the aforementioned topics illustrates the justification for providing a means for centralizing and disseminating information from the broader, but still focused, area of harmonic analysis. This will be a key role of *ANHA*. We intend to publish with the scope and interaction that such a host of issues demands.

Along with our commitment to publish mathematically significant works at the frontiers of harmonic analysis, we have a comparably strong commitment to publish major advances in the following applicable topics in which harmonic analysis plays a substantial role:

Antenna theory
Biomedical signal processing
Digital signal processing
Fast algorithms
Gabor theory and applications
Image processing
Numerical partial differential equations
Prediction theory
Radar applications
Sampling theory
Spectral estimation
Speech processing
Time-frequency and time-scale analysis
Wavelet theory

The above point of view for the *ANHA* book series is inspired by the history of Fourier analysis itself, whose tentacles reach into so many fields.

In the last two centuries, Fourier analysis has had a major impact on the development of mathematics, on the understanding of many engineering and scientific phenomena, and on the solution of some of the most important problems in mathematics and the sciences. Historically, Fourier series was developed in the analysis of some of the classical PDEs of mathematical physics; these series were used to solve such equations. In order to understand Fourier series and the kinds of solutions they could represent, some of the most basic notions of analysis were defined, e.g., the concept of "function." Since the coefficients of Fourier series are integrals, it is no surprise that Riemann integrals were conceived to deal with uniqueness properties of trigonometric series. Cantor's set theory was also developed because of such uniqueness questions.

A basic problem in Fourier analysis is to show how complicated phenomena, such as sound waves, can be described in terms of elementary harmonics. There are two aspects of this problem: first, to find, or even define properly, the harmonics or spectrum of a given phenomenon, e.g., the spectroscopy problem in optics; second, to determine which phenomena can be constructed from given classes of harmonics, as done, for example, by the mechanical synthesizers in tidal analysis.

Fourier analysis is also the natural setting for many other problems in engineering, mathematics, and the sciences. For example, Wiener's Tauberian theorem in Fourier analysis not only characterizes the behavior of the prime numbers, but also provides the proper notion of spectrum for phenomena such as white light; this latter process leads to the Fourier analysis associated with correlation functions in filtering and prediction problems, and these problems, in turn, deal naturally with Hardy spaces in the theory of complex variables.

Nowadays, some of the theory of PDEs has given way to the study of Fourier integral operators. Problems in antenna theory are studied in terms of unimodular trigonometric polynomials. Applications of Fourier analysis abound in signal processing, whether with the fast Fourier transform (FFT), or filter design, or the adaptive modeling inherent in time-frequency-scale methods such as wavelet theory. The coherent states of mathematical physics are translated and modulated Fourier transforms, and these are used, in conjunction with the uncertainty principle, for dealing with signal reconstruction in communications theory. We are back to the raison d'être of the *ANHA* series!

College Park, MD, USA

John J. Benedetto
Series Editor
University of Maryland

Preface

Compressed sensing is an efficient technique to measure and reconstruct high-dimensional signals. The key idea of this method is that high-dimensional signals usually admit a lower dimensional structure in the sense that they have a sparse representation in a basis or a frame.

Since the publication of the first papers in 2006, compressed sensing has established itself as an independent field of research, and its mathematical foundations are nowadays well understood. Along the way, the area has benefitted and was driven by its interdisciplinarity. Indeed, compressed sensing is located at the interface of applied mathematics and engineering with applications in communication theory, imaging sciences, optics, radar technology, sensor networks, and tomography.

In this spirit, two MATHEON conferences entitled "Compressed Sensing and its Applications" were held in December 2013 and December 2015 at the Technische Universität Berlin. These brought together experts from a variety of research areas including electrical engineering, mathematics, biology, chemistry, computer science, or material scientists. Both workshops were supported by the Matheon, which is a research center in Berlin for "Mathematics for Key Technologies", as well as by the German Research Foundation (DFG).

Due to the overwhelming success of the previous workshops, the editors of this volume organized a third edition of the conference series in 2017. In addition to the established field of compressed sensing, we decided to open the conference up to applications of deep learning in data science as we expected substantial overlap of these methods and ideas and those prevalent in compressed sensing. Overall, we welcomed 140 participants from 12 countries with an immense variety of different backgrounds leading to fruitful and inspiring discussions.

This volume contains a selection of contributions from speakers of this conference. It is aimed at a broad readership including graduate students and researchers in the areas of mathematics, computer science, and engineering. We believe it is also accessible to researchers working in any other field requiring methodologies for data science. Hence, this volume can be used both as a state-of-the-art monograph on applications of compressed sensing and as a textbook for graduate students. Here is a brief outline of the contents of each chapter.

Chapter "An Introduction to Compressed Sensing" provides an introduction as well as a self-contained overview of the main results on the theory and applications of compressed sensing. Chapters "Quantized Compressed Sensing: A Survey", "On Reconstructing Functions from Binary Measurements", and "Classification Scheme for Binary Data with Extensions" explore the role of quantization in data science applications. More specifically, Chapter "Quantized Compressed Sensing: A Survey" gives a survey on quantized compressed sensing, Chapter "On Reconstructing Functions from Binary Measurements" analyses reconstruction of functions from binary measurements and Chapter "Classification Scheme for Binary Data with Extensions" introduces a classification algorithm from binary measurements. Chapters "Generalization Error in Deep Learning", "Deep Learning for Trivial Inverse Problems", and "Oracle Inequalities for Local and Global Empirical Risk Minimizers" discuss aspects of the area of machine learning. To be precise, Chapter "Generalization Error in Deep Learning" presents a survey on theoretical results on the generalization error in machine learning techniques. Chapter "Deep Learning for Trivial Inverse Problems" studies the feasibility of deep learning techniques to solve inverse problems. Chapter "Oracle Inequalities for Local and Global Empirical Risk Minimizers" establishes oracle inequalities for empirical risk minimization. Chapter "Median-Truncated Gradient Descent: A Robust and Scalable Nonconvex Approach for Signal Estimation" presents a variation of gradient descent with applications in traditional compressed sensing as well as machine learning. In the final chapter of this book a practical example of compressed sensing in single pixel imaging is presented.

This conference certainly would not have been possible without the support of dedicated volunteers, and we gratefully acknowledge the help of all members of the Applied Functional Analysis Group at the Technische Universität Berlin Tiep Dovan, Katharina Eller, Axel Flinth, Ansgar Freyer, Ingo Gühring, Martin Genzel, Ali Hashemi, Anja Hedrich, Sandra Keiper, Héctor Andrade Loarca, Jan Macdonald, and Stephan Wäldchen.

Munich, Germany	Holger Boche
Durham, USA	Robert Calderbank
Berlin, Germany	Giuseppe Caire
Berlin, Germany	Gitta Kutyniok
Aachen, Germany	Rudolf Mathar
Oxford, UK	Philipp Petersen
April 2019	

Contents

An Introduction to Compressed Sensing 1
Niklas Koep, Arash Behboodi and Rudolf Mathar
1 Introduction 2
2 Preliminaries 4
2.1 Norms and Quasinorms 5
2.2 Random Variables, Vectors, and Matrices 7
3 Signal Models 10
3.1 Sparse Vectors 12
3.2 Block- and Group-Sparse Vectors 16
3.3 Low-Rank Matrices 17
3.4 Low-Complexity Models in Bases and Frames 20
4 Recovery of Individual Vectors 21
4.1 Exact Recovery 21
4.2 Connections to Conic Integral Geometry 28
5 Exact Recovery of Sparse Vectors 32
6 Characterization of Measurement Matrices 34
6.1 Null Space Property 35
6.2 Restricted Isometry Property 38
6.3 Mutual Coherence 39
6.4 Quotient Property 40
7 Probabilistic Constructions of Measurement Matrices 41
7.1 Restricted Isometries 41
7.2 Random Matrices and the Null Space Property 45
8 An Algorithmic Primer 47
8.1 Convex Programming 48
8.2 Thresholding Algorithms 49
8.3 Greedy Methods 55
8.4 Iteratively Reweighted Least-Squares 58

9 Conclusion 60
References 61

Quantized Compressed Sensing: A Survey 67
Sjoerd Dirksen
1 Introduction 67
1.1 Notation 68
2 Key Concepts 69
3 Two Fundamental Limits 72
4 One-Bit Compressed Sensing 74
4.1 Memoryless One-Bit Compressed Sensing: Zero Thresholds 74
4.2 Memoryless One-Bit Compressed Sensing With Dithering 78
4.3 Relation to Binary Embeddings 82
4.4 Exponential Error Decay Via Adaptive Thresholds 82
5 Memoryless Multi-bit Compressed Sensing 85
6 Noise-Shaping Methods 89
References 92

On Reconstructing Functions from Binary Measurements 97
Robert Calderbank, Anders Hansen, Bogdan Roman and Laura Thesing
1 Introduction 98
2 Walsh Functions 101
2.1 Definition 101
2.2 Properties 104
3 Reconstruction Space 105
4 Linear Reconstruction Methods 107
4.1 PBDW Method 107
4.2 Generalized Sampling 108
4.3 The Stable Sampling Rate for the Walsh-Wavelet Case 110
5 Non-linear Reconstruction Methods 112
5.1 Classical Compressed Sensing 112
5.2 Types of Compressed Sensing Problems 114
5.3 Taking Structure and Infinite Dimensionality into Account 114
5.4 Points of Discussion Regarding Structure 121
6 Conclusion 125
References 126

Classification Scheme for Binary Data with Extensions 129
Denali Molitor, Deanna Needell, Aaron Nelson, Rayan Saab and Palina Salanevich
1 Introduction 129

1.1 Notation and Setup 130
1.2 Related Work and Background 131
1.3 Organization 132
2 Simple Classification Approach 132
2.1 Analytical Justification 135
2.2 Experimental Results 137
3 Hierarchical Classification 141
3.1 Experimental Results 142
4 Implementation Considerations 144
4.1 Parameter Selection 144
4.2 Dynamic Hyperplane Selection 147
4.3 Efficient Representations 147
5 Conclusion 148
References 149

Generalization Error in Deep Learning 153
Daniel Jakubovitz, Raja Giryes and Miguel R. D. Rodrigues
1 Introduction 153
2 The Learning Problem 155
3 Deep Neural Networks 161
4 Generalization Error in Deep Learning 165
4.1 Understanding Deep Learning Requires Rethinking Generalization 165
4.2 Exploring Generalization in Deep Learning 167
4.3 A PAC-Bayesian Approach to Spectrally Normalized Margin Bounds for Neural Networks 170
4.4 Stability and Generalization 172
4.5 Robustness and Generalization 175
4.6 Stronger Generalization Bounds for Deep Nets via a Compression Approach 178
4.7 Train Faster, Generalize Better: Stability of Stochastic Gradient Descent 179
4.8 On Large Batch Training for Deep Learning: Generalization Gap and Sharp Minima 181
4.9 Sharp Minima Solutions to the Training of DNNs Can Generalize for Deep Nets 183
4.10 Train Longer, Generalize Better: Closing the Generalization Gap in Large Batch Training of Neural Networks 183
4.11 Generalization Error of Invariant Classifiers 185
4.12 Generalization Error and Adversarial Attacks 185
5 Open Problems 186
5.1 Problem 1: Generalization and Memorization 186
5.2 Problem 2: Generalization and Robustness 187

5.3 Problem 3: Generalization and Adversarial Examples 188
5.4 Problem 4: Generalization Error of Generative Models 188
5.5 Problem 5: Generalization Error and the Information Bottleneck 189
6 Conclusions 189
References 190

Deep Learning for Trivial Inverse Problems 195
Peter Maass
1 Motivation and Outline 195
2 Basic Example 196
2.1 Testing Error Convergence for Various Values of ε 199
3 Analysis of Trivial Neural Networks for Inverse Problems 200
3.1 Matrix Case 201
4 Further Numerical Tests 207
References 209

Oracle Inequalities for Local and Global Empirical Risk Minimizers 211
Andreas Elsener and Sara van de Geer
1 Introduction 211
1.1 Model Misspecification 211
1.2 Penalized Empirical Risk Minimization 212
1.3 Conditions on the Risk 214
1.4 Norm Penalties 215
1.5 Related Literature 217
1.6 Organization 218
2 Sharp Oracle Inequalities 218
2.1 Convex and Nondifferentiable 219
2.2 Nonconvex and Differentiable 220
2.3 Asymptotic Interpretation 220
3 Applications 221
3.1 Regression 221
3.2 Classification 227
3.3 Dimension Reduction 228
4 Discussion 231
5 Appendix 232
References 235

Median-Truncated Gradient Descent: A Robust and Scalable Nonconvex Approach for Signal Estimation 237
Yuejie Chi, Yuanxin Li, Huishuai Zhang and Yingbin Liang
1 Introduction 238
2 Median-Truncated Gradient Descent 239

3 Robust Low-Rank Matrix Recovery 241
3.1 Median-TGD for Robust Low-Rank Matrix Recovery 242
3.2 Theoretical Guarantees 243
4 Robust Phase Retrieval 245
4.1 Median-TGD for Robust Phase Retrieval 246
4.2 Performance Guarantees 247
5 Highlights of Theoretical Analysis 248
5.1 Useful Properties of Median 249
5.2 Regularity Condition 250
6 Numerical Experiments 251
6.1 Median-TGD for Low-Rank Matrix Recovery 251
6.2 Median-TGD for Phase Retrieval 252
7 Related Works 257
8 Conclusion 258
References 258

Reconstruction Methods in THz Single-Pixel Imaging 263
Martin Burger, Lea Föcke, Lukas Nickel, Peter Jung and Sven Augustin
1 Introduction 263
2 Compressed Sensing and Reconstruction in Single-Pixel Cameras 265
2.1 Compressed Sensing Techniques 265
2.2 Total Variation Regularization and Related Methods 267
3 Computational Image Reconstruction 269
4 Challenges in Practical THz Single-Pixel Imaging 271
5 Calibration Problems 272
5.1 Self-calibration and Bilinear Inverse Problems 273
5.2 Single-Pixel and Retinex Denoising 274
6 Phase Retrieval in Single-Pixel Cameras 278
6.1 Algorithms for Phase Retrieval 279
6.2 Results 282
7 Conclusion 287
References 288

Applied and Numerical Harmonic Analysis 291

Acronyms

ADC	Analog-to-digital converter
ADMM	Alternating direction method of multipliers
AMP	Approximate message passing
BOS	Bounded orthonormal system
BP	Basis pursuit
BPDN	Basis pursuit denoising
CoSaMP	Compressive sampling matching pursuit
CS	Compressed sensing
DCT	Discrete cosine transform
DFT	Discrete Fourier transform
EM	Expectation maximization
FFT	Fast Fourier transform
FISTA	Fast iterative soft thresholding algorithm
HTP	Hard thresholding pursuit
IHT	Iterative hard thresholding
IRLS	Iteratively reweighted least squares
ISTA	Iterative soft thresholding algorithm
LASSO	Least-absolute shrinkage selection operator
LP	Linear program
MGF	Moment generating function
MSE	Mean squared error
NSP	Null space property
OMP	Orthogonal matching pursuit
QCBP	Quadratically constrained basis pursuit
RIC	Restricted isometry constant
RIP	Restricted isometry property
SOCP	Second-order cone program
SOS	Sum of squares

An Introduction to Compressed Sensing

Niklas Koep, Arash Behboodi and Rudolf Mathar

Abstract Compressed sensing and many research activities associated with it can be seen as a framework for signal processing of low-complexity structures. A cornerstone of the underlying theory is the study of inverse problems with linear or nonlinear measurements. Whether it is sparsity, low-rankness, or other familiar notions of low complexity, the theory addresses necessary and sufficient conditions behind the measurement process to guarantee signal reconstruction with efficient algorithms. This includes consideration of robustness to measurement noise and stability with respect to signal model inaccuracies. This introduction aims to provide an overall view of some of the most important results in this direction. After discussing various examples of low-complexity signal models, two approaches to linear inverse problems are introduced which, respectively, focus on the recovery of individual signals and recovery of all low-complexity signals simultaneously. In particular, we focus on the former setting, giving rise to so-called nonuniform signal recovery problems. We discuss different necessary and sufficient conditions for stable and robust signal reconstruction using convex optimization methods. Appealing to concepts from non-asymptotic random matrix theory, we outline how certain classes of random sensing matrices, which fully govern the measurement process, satisfy certain sufficient conditions for signal recovery. Finally, we review some of the most prominent algorithms for signal recovery proposed in the literature.

N. Koep · A. Behboodi (✉) · R. Mathar
RWTH Aachen Theoretische Informationstechnik, 52056 Aachen, Germany
e-mail: arash.behboodi@ti.rwth-aachen.de

N. Koep
e-mail: niklas.koep@ti.rwth-aachen.de

R. Mathar
e-mail: rudolf.mathar@ti.rwth-aachen.de

H. Boche et al. (eds.), *Compressed Sensing and Its Applications*,
Applied and Numerical Harmonic Analysis,
https://doi.org/10.1007/978-3-319-73074-5_1

1 Introduction

The field of compressed sensing was originally established with the publication of the seminal papers "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information" [25] by Terence Tao, Justin Romberg and Emmanuel Candès, and the aptly titled "Compressed sensing" [40] by David Donoho. The research activity by hundreds of researchers that followed over time transformed the field into a mature mathematical theory with far-reaching implications in applied mathematics and engineering alike. While deemed impossible by the celebrated Shannon–Nyquist sampling theorem, as well as fundamental facts in linear algebra, their work demonstrated that unique solutions of underdetermined systems of linear equations do in fact exist if one limits attention to signal sets exhibiting some type of low-complexity structure. In particular, Tao, Romberg, Candès, and Donoho considered so-called sparse vectors containing only a limited number of nonzero coefficients and demonstrated that solving a simple linear program minimizing the ℓ_1-norm of a vector subject to an affine constraint allowed for an efficient way to recover such signals. While examples of ℓ_1-regularized methods as a means to retrieve sparse estimates of linear inverse problems can be traced back as far as the 1970s to work in seismology, the concept was first put on a rigorous footing in a series of landmark papers [25–28, 40]. Today, compressed sensing is considered a mature field firmly positioned at the intersection of linear algebra, probability theory, convex analysis, and Banach space theory.

This chapter serves as a concise overview of the field of compressed sensing, highlighting some of the most important results in the theory, as well as some more recent developments. In light of the popularity of the field, there truly exists no shortage of excellent surveys and introductions to the topic. We want to point out the following references in particular: [14, 46, 47, 51, 52, 54], which include extended monographs focusing on a rigorous presentation of the mathematical theory, as well as works more focused on the application side, e.g., in the context of wireless communication [60] or more generally in sparse signal processing [29]. Due to the volume of excellent references, we decided on a rather opinionated selection of topics for this introduction. For instance, a notable omission of our text is a discussion on the so-called Gelfand widths, a concept in the theory of Banach spaces that is commonly used in compressed sensing to prove the optimality of bounds on the number of measurements required to establish certain properties of random matrices. Moreover, in the interest of space, we opted to omit most of the proofs in this chapter, and instead make frequent reference to the excellent material found in the literature.

Organization

Given the typical syllabus of introductions to compressed sensing, we decided to go a slightly different route than usual by motivating the underlying problem from an extended view at the problem of individual vector recovery before moving on to the so-called uniform recovery case which deals with the simultaneous recovery of all vectors in a particular signal class at once.

In Sect. 2, we briefly recall a few basic definitions of norms and random variables. We also define some basic notions about so-called *subgaussian* random variables as they play a particularly important role in modern treatments of compressed sensing.

In Sect. 3, we introduce a variety of signal models for different applications and contexts. To that end, we adopt the notion of *simple sets* generated by so-called *atomic sets*, and the associated concept of *atomic norms* which provide a convenient abstraction for the formulation of nonuniform recovery problems in a multitude of different domains. In the context of sparse recovery, we also discuss the important class of so-called *compressible vectors* as a practical alternative to exactly sparse vectors to model real-world signals such as natural images, audio signals, and the like.

Equipped with the concept of the atomic norm which gives rise to a tractable recovery program of central importance in the context of linear inverse problems, we discuss in Sect. 4 conditions for perfect or robust recovery of low-complexity signals. We also comment on a rather recent development in the theory which connects the problem of sparse recovery with the field of conic integral geometry.

Starting with Sect. 5, we finally turn our attention to the important case of uniform recovery of sparse or compressible vectors where we are interested in establishing guarantees which—given a particular measurement matrix—hold uniformly over the entire signal class. Such results stand in stark contrast to the problems we discuss in Sect. 4 where recovery conditions are allowed to locally depend on the choice of the particular vector one aims to recover.

In Sect. 6, we introduce a variety of properties of sensing matrices such as the null space property and the restricted isometry property which are commonly used to assert that recovery conditions as teased in Sect. 5 hold for a particular matrix. While the deterministic construction of matrices with provably optimal number of measurements remains a yet unsolved problem, random matrices—including a broad class of structured random matrices—which satisfy said properties can be shown to exist in abundance. We therefore complement our discussion with an overview of some of the most important classes of random matrices considered in compressed sensing in Sect. 7.

We conclude our introduction to the field of compressed sensing with a short survey of some of the most important sparse recovery algorithms in Sect. 8.

Motivation

At the heart of compressed sensing (CS) lies a very simple question. Given a d-dimensional vector $\mathring{\mathbf{x}}$, and a set of m measurements of the form $y_i = \langle \mathbf{a}_i, \mathring{\mathbf{x}} \rangle$, under what conditions are we able to infer $\mathring{\mathbf{x}}$ from knowledge of

$$\mathbf{A} = (\mathbf{a}_1, \ldots, \mathbf{a}_m)^\top \quad \text{and} \quad \mathbf{y} = (y_1, \ldots, y_m)^\top$$

alone? Historically, the answer to this question was "as soon as $m \geq d$" or more precisely, as soon as $\operatorname{rank}(\mathbf{A}) = d$. In other words, the number of independent observations of $\mathring{\mathbf{x}}$ has to exceed the number of unknowns in $\mathring{\mathbf{x}}$, namely, the dimension of the vector space V containing it. The beautiful insight of compressed sensing is that this statement is actually too pessimistic if the information content in $\mathring{\mathbf{x}}$ is less than d. The only exception to this rule that was known prior to the inception of the field of compressed sensing was when $\mathring{\mathbf{x}}$ was known to live in a lower dimensional linear subspace $W \subset V$ with $\dim(W) \leq d$. A highly oversimplified summary of the contribution of compressed sensing therefore says that the field extended the previous observation from single subspaces to unions of subspaces. This interpretation of the set of sparse vectors is therefore also known as the *union-of-subspaces* model. While sparsity is certainly firmly positioned at the forefront of CS research, the concept of low-complexity models encompasses many other interesting structures such as block- or group-sparsity, as well as low-rankness of matrices to name a few.

We will comment on such signal models in Sect. 3. As hinted at before, the recovery of these signal classes can be treated in a unified way using the atomic norm formalism (cf. Sect. 4) as long as we are only interested in nonuniform recovery results. Establishing similar results which hold uniformly over entire signal classes, however, usually requires more specialized analyses. In the later parts of this introduction, we therefore limit our discussions to sparse vectors. Note that while more restrictive low-complexity structures such as block- or group-sparsity overlap with the class of sparse vectors, the recovery guarantees obtained by merely modeling such signals as sparse are generally suboptimal as they do not exploit all latent structure inherent to their respective class.

Before moving on to a more detailed discussion of the most common signal models, we briefly want to comment on a particular line of research that deals with low-complexity signal recovery from nonlinear observations. Consider an arbitrary univariate, scalar-valued function f acting element-wise on vectors:

$$\mathbf{y} = f(\mathbf{A}\mathbf{x}). \tag{1}$$

An interesting instance of Eq. (1) is when f models the effects of an analog-to-digital converter (ADC), mapping the infinite-precision observations $\mathbf{A}\mathbf{x}$ on a finite quantization alphabet. Since this extension of the linear observation model gives rise to its very own set of problems which require specialized tools beyond what is needed in the basic theory of compressed sensing, we will not discuss this particular measurement paradigm in this introduction. A good introduction to the general topic of nonlinear signal recovery can be found in [100]. For a detailed survey focusing on the comparatively young field of quantized compressed sensing, we refer interested readers to [16].

2 Preliminaries

Compressed sensing builds on various mathematical tools from linear algebra, optimization theory, probability theory, and geometric functional analysis. In this section, we review some of the mathematical notions used throughout this chapter. We start with a few remarks regarding notation.

Notation

We use lower- and uppercase boldface letters to denote vectors and matrices, respectively. The all ones vector of appropriate dimension is denoted by $\mathbf{1}$, the zero vector is $\mathbf{0}$, and the identity matrix is Id. Given a natural number $n \in \mathbb{N}$, we denote by $[n]$ the set of integers from 1 to n, i.e., $[n] := \{1, \ldots, n\} = \mathbb{N} \cap [1, n]$. The complement of a subset $A \subset B$ is denoted by $\overline{A} = B \backslash A$. For a vector $\mathbf{x} \in \mathbb{C}^d$ and an index set $S \subset [d]$ with $|S| = k$, the meaning of $\mathbf{x}_S$ may change slightly depending on context. In particular, it might denote the vector $\mathbf{x}_S \in \mathbb{C}^d$ which agrees with $\mathbf{x}$ only on the index set S, and vanishes identically otherwise. On the other hand, it might represent the k-dimensional vector restricted to the coordinates indexed by S. The particular meaning should be apparent from context. Finally, for $a, b > 0$, the notation $a \lesssim b$ hides an absolute constant $C > 0$, which does not depend on either a or b, such that $a \leq Cb$ holds.

2.1 Norms and Quasinorms

The vectors we consider in this chapter are generally assumed to belong to a finite- or infinite-dimensional Hilbert space $\mathcal{H}$, i.e., a vector space endowed with a bilinear form $\langle \cdot, \cdot \rangle : \mathcal{H} \times \mathcal{H} \to \mathbb{R}$ known as *inner product*, which induces a norm on the underlying vector space by[1]

$$\|\mathbf{x}\| := \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle}.$$

The d-dimensional Euclidean space $\mathbb{R}^d$ is an example of a vector space with the inner product between $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ defined as

$$\langle \mathbf{x}, \mathbf{y} \rangle := \mathbf{x}^\top \mathbf{y} = \sum_{i=1}^{d} x_i y_i.$$

The norm induced by this inner product corresponds to the so-called ℓ_2-norm. In general, the family of ℓ_p-norms on $\mathbb{R}^d$ is defined as

$$\|\mathbf{x}\|_p := \begin{cases} \left(\sum_{i=1}^d |x_i|^p \right)^{1/p}, & p \in [1, \infty) \\ \max_{i \in [d]} |x_i|, & p = \infty. \end{cases}$$

Note that the ℓ_2-norm is the only ℓ_p-norm on $\mathbb{R}^d$ that is induced by an inner product since it satisfies the parallelogram identity. One can extend the definition of ℓ_p-norms to the case $p \in (0, 1)$. However, the resulting "ℓ_p-norm" ceases to be a norm as it no longer satisfies the triangle inequality. Instead, the collection of ℓ_p-norms for $p \in (0, 1)$ defines a family of quasinorms which satisfy the weaker condition

[1] Technically, a Hilbert space is an inner product space in which every Cauchy sequence converges to a point in the same space.

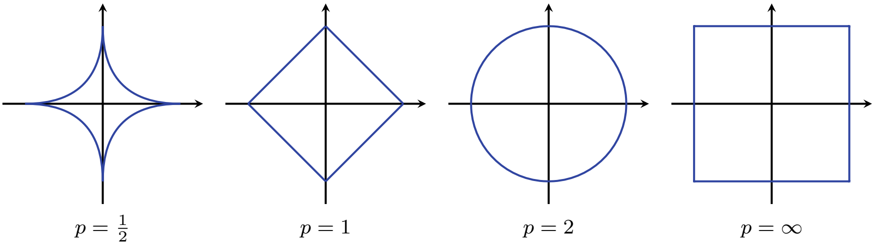


Fig. 1 The ℓ_p-unit spheres in $\mathbb{R}^2$ for different values of p. The interiors (including their respective boundaries) correspond to the ℓ_p-balls $\mathbb{B}_p^d$

$$\|\mathbf{x} + \mathbf{y}\|_p \leq 2^{1/p-1}(\|\mathbf{x}\|_p + \|\mathbf{y}\|_p).$$

Additionally, we will make frequent use of the egregiously termed ℓ_0-norm of $\mathbf{x}$ which is defined as the number of nonzero coefficients,

$$\|\mathbf{x}\|_0 := \lim_{p \to 0} \|\mathbf{x}\|_p^p = |\mathrm{supp}(\mathbf{x})| \,.$$

Note that the ℓ_0-norm, as a measure of *sparsity* of a vector, is neither a norm nor a quasinorm (or even a seminorm) as it is not positively homogeneous, i.e., for $t > 0$ we have $\|t\mathbf{x}\|_0 = \|\mathbf{x}\|_0 \neq t \, \|\mathbf{x}\|_0$. As we will see later, both the ℓ_1-norm, and the ℓ_p-quasinorms are of particular interest in the theory of compressed sensing. The ℓ_p-unit ball, defined as

$$\mathbb{B}_p^d := \left\{\mathbf{x} \in \mathbb{C}^d : \|\mathbf{x}\|_p \leq 1\right\},$$

forms a convex body for $p \geq 1$ and a nonconvex one for $p \in (0, 1)$. The boundaries $\partial\mathbb{B}_p^d = \{\mathbf{x} : \|\mathbf{x}\|_p = 1\}$ correspond to the ℓ_p-unit spheres. For $p = 2$, the boundary $\partial\mathbb{B}_2^d$ of the ℓ_2-ball corresponds to the unit Euclidean sphere denoted $\mathbb{S}^{d-1}$. Some examples of the ℓ_p-unit spheres are given in Fig. 1.

Another commonly used space in compressed sensing is the space of linear transformations from $\mathbb{R}^d$ to $\mathbb{R}^m$. This particular function space is isomorphic to the collection of $\mathbb{R}^{m \times d}$ matrices and forms a vector space on which we can define an inner product via

$$\langle \mathbf{A}, \mathbf{B} \rangle := \mathrm{tr}(\mathbf{A}^\top \mathbf{B}).$$

The norm induced by this inner product is called the Frobenius norm and is given by

$$\|\mathbf{A}\|_{\mathrm{F}} := \sqrt{\mathrm{tr}(\mathbf{A}^\top \mathbf{A})} = \sqrt{\sum_{i \in [d]} \sum_{j \in [m]} a_{ij}^2}.$$

In this context, the inner product above is also known as the so-called *Frobenius* inner product. Another commonly used norm defined on the space of linear transformations is the operator norm

$$\|\mathbf{A}\|_{p\to q} := \sup_{\|\mathbf{x}\|_p \le 1} \|\mathbf{A}\mathbf{x}\|_q .$$

In particular, the operator norm $\|\mathbf{A}\|_{2\to 2}$ between two normed spaces equipped with their respective ℓ_2-norm is given by the maximum singular value of $\mathbf{A}$ denoted by $\sigma_{\max}(\mathbf{A})$.

2.2 Random Variables, Vectors, and Matrices

Let $(\boldsymbol{\Omega}, \Sigma, \mathbb{P})$ be a probability space consisting of the sample space $\boldsymbol{\Omega}$, the Borel measurable event space Σ, and a probability measure $\mathbb{P}\colon \Sigma \to [0, 1]$. The space of matrix-valued, Borel measurable functions from $\boldsymbol{\Omega}$ to $\mathbb{R}^{m\times d}$ are called *random matrices*. This space inherits a probability measure as the pushforward of the measure $\mathbb{P}$. For $d = 1$, we obtain the set of random vectors; the space of random variables corresponds to the choice $m = d = 1$. Given a scalar random variable X, the *expected value* of X is defined as

$$\mathbb{E}X := \int X \mathrm{d}\mathbb{P} = \int_{\boldsymbol{\Omega}} X(\omega)\mathrm{d}\mathbb{P}(\omega)$$

if the integral exists. Moreover, if $\mathbb{E}e^{tX}$ exists for all $|t| < h$ for some $h \in \mathbb{R}$, then the map

$$M_X\colon \mathbb{R} \to \mathbb{R}\colon t \mapsto M_X(t) = \mathbb{E}e^{tX} = \int e^{tX}\mathrm{d}\mathbb{P},$$

known as the moment generating function (MGF), fully determines the distribution of X. The pth absolute moment of a random variable X is defined as

$$\mathbb{E}|X|^p = \int_{\boldsymbol{\Omega}} |X(\omega)|^p \mathrm{d}\mathbb{P}(\omega).$$

This leads to the notion of the so-called L^p norm

$$\|X\|_{L^p} := (\mathbb{E}|X|^p)^{1/p},$$

which turns the space of random variables equipped with $\|\cdot\|_{L^p}$ into a normed vector space. A particular class of random variables which finds widespread use in compressed sensing is the so-called subgaussian random variables whose L^p norm increases at most as $\sqrt{p}$. The name *subgaussian* is owed to the fact that subgaussian random variables have tail probabilities which decay at least as fast as the tails of the Gaussian distribution [99]. This leads to the following definition.

Definition 1 (*Subgaussian random variables*) A random variable X is called subgaussian if it satisfies one of the following equivalent properties:

1. The tails of X satisfy
$$\mathbb{P}(|X| \geq t) \leq 2\exp(-t^2/K_1^2) \quad t \geq 0.$$
2. The absolute moments of X satisfy
$$(\mathbb{E}|X|^p)^{1/p} \leq K_2\sqrt{p} \quad \forall p \geq 1.$$
3. The super-exponential moment of X satisfies
$$\mathbb{E}\exp(X^2/K_3^2) \leq 2.$$
4. If $\mathbb{E}X = 0$, then the MGF of S satisfies
$$\mathbb{E}\exp(tX) \leq \exp(K_4^2 t^2) \quad \forall t \in \mathbb{R}.$$

The constants $K_1, \ldots, K_4$ are universal.

Note that the constants $K_i > 0$ for $i = 1, 2, 3, 4$ differ from each other by at most a constant factor, which, in turn, deviate only by a constant factor from the so-called subgaussian norm $\|\cdot\|_{\psi_2}$.

Definition 2 (*Subgaussian norm*) Given a random variable X, we define the subgaussian norm of X as

$$\|X\|_{\psi_2} := \inf\{s > 0 : \mathbb{E}\psi_2(X/s) \leq 1\},$$

where $\psi_2(t) := \exp(t^2) - 1$ is called an *Orlicz function*.

The set of subgaussian random variables defined on a common probability space equipped with the norm $\|\cdot\|_{\psi_2}$ therefore forms a normed space known as *Orlicz space*. Note that some authors instead define the subgaussian norm as

$$\|X\|_{\psi_2} := \sup_{p\geq 1} \frac{1}{\sqrt{p}} (\mathbb{E}|X|^p)^{1/p}. \tag{2}$$

In light of Definition 1, these definitions are equivalent up to a multiplicative constant. As a consequence of Eq. (2) and Definition 2 above, a random variable is subgaussian if its subgaussian norm is finite. For instance, the subgaussian norm of a Gaussian random variable $X \sim \mathsf{N}(0, \sigma^2)$ is—up to a constant—multiplicatively bounded from above by σ. The subgaussian norm of a Rademacher random variable is given by $\|X\|_{\psi_2} = 1/\sqrt{\log 2}$. Gaussian and Bernoulli random variables are therefore typical instances of subgaussian random variables. Other examples include random variables following the Steinhaus[2] distribution, as well as any bounded random variables in general.

A convenient property of subgaussian random variables is that their tail probabilities can be expressed in terms of their subgaussian norm:

$$\mathbb{P}(|X| \geq t) \leq 2\exp\left(-\frac{ct^2}{\|X\|_{\psi_2}^2}\right) \quad \forall t > 0.$$

If $X_i \sim \mathsf{N}(0, \sigma_i^2)$ are independent Gaussian random variables, then due to the rotation invariance of the normal distribution, the linear combination $X = \sum_i X_i$ is still a zero-mean Gaussian random variable with variance $\sum_i \sigma_i^2$. This property also extends to subgaussians barring a dependence a multiplicative constant, i.e., if $(X_i)_i$ is a sequence of centered subgaussian random variables, then

$$\|\sum_i X_i\|_{\psi_2}^2 \leq C\sum_i \|X_i\|_{\psi_2}^2.$$

This can easily be shown with the help of the moment generating function of $X = \sum_i X_i$. The rotational invariance along with the tail property of subgaussian distributions makes it possible to generalize many familiar tools such as Hoeffding-type inequalities to subgaussian distributions, e.g.,

$$\mathbb{P}\left(\left|\sum_i X_i\right| \geq t\right) \leq 2\exp\left(-\frac{ct^2}{\sum_i \|X_i\|_{\psi_2}^2}\right) \quad \forall t > 0.$$

Oftentimes, it is convenient to extend the notion of subgaussianity from random variables to random vectors. In particular, we say that a random vector $\mathbf{X} \in \mathbb{R}^m$ is subgaussian if the random variable $X = \langle \mathbf{X}, \mathbf{y} \rangle$ is subgaussian for all $\mathbf{y} \in \mathbb{R}^m$. Taking the supremum of the subgaussian norm of X over all unit directions then leads to the definition of the subgaussian norm for random vectors.

Definition 3 (*Subgaussian vector norm*) The subgaussian norm of an m-dimensional random vector $\mathbf{X}$ is

$$\|\mathbf{X}\|_{\psi_2} := \sup_{\mathbf{y}\in\mathbb{S}^{m-1}} \| \langle \mathbf{X}, \mathbf{y} \rangle \|_{\psi_2}.$$

Finally, a random vector $\mathbf{X}$ is called isotropic if $\mathbb{E}| \langle \mathbf{X}, \mathbf{y} \rangle |^2 = \|\mathbf{y}\|_2^2$ for all $\mathbf{y} \in \mathbb{R}^m$.

[2] A Steinhaus random variable is a complex random variable distributed uniformly on the complex unit circle.

3 Signal Models

As a basic framework for the types of signals discussed in this introduction, we decided to adopt the notion of so-called *atomic sets* as coined by Chandrasekaran et al. [31]. This serves two purposes. First, it elegantly emphasizes the notion of *low complexity* of the signals one aims to recover or estimate in practice. Second, the associated notion of *atomic norm* (cf. Definition 5) provides a convenient way to motivate certain geometric ideas in the recovery of low-complexity models. Let us emphasize that this viewpoint is not necessarily required when discussing so-called uniform recovery results where one is interested in conditions allowing for the recovery of entire signal classes given a fixed draw of a measurement matrix (cf. Sect. 7). However, the concept provides a suitable level of abstraction to discuss recovery conditions for individual vectors of a variety of different interesting signal models in a unified manner which were previously studied in isolation by researchers in their respective fields.

As alluded to in the motivation, one of the most common examples of a "low-complexity" structure of a signal $\mathring{\mathbf{x}} \in \mathbb{C}^d$ is the assumption that it belongs to a lower dimensional subspace of dimension k. Given a matrix $\mathbf{U} \in \mathbb{C}^{d \times k}$ whose columns $\mathbf{u}_i$ span said subspace, and the linear measurements $\mathbf{y} = \mathbf{A}\mathbf{x}$, we may simply solve the least-squares problem

$$\underset{\mathbf{c} \in \mathbb{C}^k}{\text{minimize}} \quad \|\mathbf{y} - \mathbf{A}\mathbf{U}\mathbf{c}\|_2 \tag{3}$$

to recover $\mathring{\mathbf{x}} = \mathbf{U}\mathbf{c}^\star$ where the solution $\mathbf{c}^\star$ of Problem (3) admits a closed-form expression in terms of the Moore–Penrose pseudoinverse of $\mathbf{AU}$. Once again, this strategy succeeds if $m \geq \dim \operatorname{span}(\{\mathbf{u}_i\}_{i=1}^k)$, i.e., if we obtain at least as many measurements as the subspace dimension. As a canonical example, assume that $\mathbf{U}$ corresponds to the identity matrix Id restricted to the columns indexed by a set $S \subset [d]$ of cardinality $|S| = k$, i.e., $\mathbf{U} = \mathrm{Id}_S$. The columns of this matrix form a basis for a k-dimensional coordinate subspace of $\mathbb{C}^d$. If we lift the restriction that $\mathring{\mathbf{x}}$ lives in this particular subspace, and rather assume instead that $\mathring{\mathbf{x}}$ belongs to any of the $\binom{d}{k}$ coordinate subspaces of dimension k, we arrive at a special case of the so-called *union-of-subspaces* model. In particular, we have

$$\mathring{\mathbf{x}} \in \bigcup_{\substack{S \subset [d], \\ |S|=k}} W_S =: \Sigma_k,$$

where W_S denotes the coordinate subspace of $\mathbb{C}^d$ with basis matrix Id_S. The set Σ_k therefore corresponds to the set of sparse vectors supported on an index set S of cardinality at most k. This signal class represents a central object of study in the field of compressed sensing.

Equipped with the knowledge that $\mathring{\mathbf{x}}$ lives in one of the k-dimensional coordinate subspaces, one could attempt to recover $\mathring{\mathbf{x}}$ by solving Problem (3) for each W_S

independently. However, even though the true solution $\mathring{\mathbf{x}}$ must be among these least-squares solutions, there is no way for us to identify the correct one. Moreover, even for moderately sized problems, the number $\binom{d}{k}$ of least-squares projections one needs to solve becomes unreasonably high. On the other hand, ignoring the information that $\mathring{\mathbf{x}}$ lives in k-dimensional subspace, and instead solving the least-squares minimization problem

$$\underset{\mathbf{x}\in\mathbb{C}^d}{\text{minimize}} \quad \|\mathbf{y}-\mathbf{A}\mathbf{x}\|_2$$

will not help either since the ℓ_2-norm we are minimizing tends to spread the signal energy over the entire support of the minimizer $\mathbf{x}^\star$ (see, e.g., the discussion in [18, Sect. 6.1.2]). We will discuss in Sect. 5 that all these issues can be resolved by imposing certain structural constraints on the measurement matrix $\mathbf{A}$, and replacing the optimization problem (3) with one that explicitly promotes the structure inherent in $\mathring{\mathbf{x}}$.

We will come back to the sparse signal model shortly. First, however, let us introduce a more flexible notion of low-complexity structures which will allow us to talk about recovery problems of more general signal models in a unified framework. As outlined above, if $\mathcal{K}$ denotes a k-dimensional subspace, then every vector in $\mathcal{K}$ can be represented as a sum of k basis vectors. To capture a similar notion of dimensionality for more general sets which do not necessarily form a subspace, we may assume that every vector in $\mathcal{K}$ can at least be represented as a linear combination of a limited number of elements in a more general generating set. While a finite-dimensional subspace is always fully determined by a finite collection of basis vectors, we now lift this finiteness requirement. The signal models generated in this fashion are simply referred to as *simple sets*.

Definition 4 (*Simple set*) Let $\mathcal{A} \subset \mathbb{C}^d$ be an origin-symmetric set whose convex hull forms a convex body.[3] Let $k \in \mathbb{N}$. Then the set

$$\mathcal{K} = \operatorname{cone}_k(\mathcal{A}) := \left\{ \mathbf{x} = \sum_{i=1}^{k} c_i \mathbf{a}_i \in \mathbb{C}^d : c_i \geq 0, \mathbf{a}_i \in \mathcal{A} \right\} \tag{4}$$

is called a *simple set*. Since $\mathcal{K}$ is generated by the set $\mathcal{A}$, we call $\mathcal{A}$ an *atomic set*.

We will discuss how this notion of simplicity leads to many familiar models in the literature on linear inverse problems. As a canonical example, however, consider the case $\mathcal{A} = \{\pm\mathbf{e}_i\} \subset \mathbb{R}^d$. The simple set $\mathcal{K}$ generated by $\operatorname{cone}_k(\mathcal{A})$ then corresponds to the set $\Sigma_k(\mathbb{R}^d)$ of k-sparse vectors.

Given an atomic set $\mathcal{A}$, we associate with it the following object.

Definition 5 (*Atomic norm*) The function

[3] A convex body is a compact convex set with non-empty interior.

$$\|\mathbf{x}\|_{\mathcal{A}} := \inf \left\{ \sum_{\mathbf{a} \in \mathcal{A}} c_{\mathbf{a}} : \mathbf{x} = \sum_{\mathbf{a} \in \mathcal{A}} c_{\mathbf{a}} \mathbf{a}, \, c_{\mathbf{a}} \geq 0 \; \forall \mathbf{a} \in \mathcal{A} \right\}$$

associated with an atomic set $\mathcal{A} \subset \mathbb{C}^d$ is called the *atomic norm* of $\mathcal{A}$ at $\mathbf{x}$.

This definition corresponds to the so-called *Minkowski functional* or *gauge* of the set conv$(\mathcal{A})$ [88, Chap. 15],

$$\gamma_{\operatorname{conv}(\mathcal{A})}(\mathbf{x}) := \inf \{ t > 0 : \mathbf{x} \in t \operatorname{conv}(\mathcal{A}) \} = \|\mathbf{x}\|_{\mathcal{A}} .$$

The norm notation $\|\cdot\|_{\mathcal{A}}$ is justified here since we assumed $\mathcal{A}$ to be compact and centrally symmetric with conv$(\mathcal{A})$ having non-empty interior. This ensures that conv$(\mathcal{A})$ is a symmetric convex body which contains an open set around the origin in which case $\|\cdot\|_{\mathcal{A}} = \gamma_{\operatorname{conv}(\mathcal{A})}(\cdot)$ defines a norm on $\mathbb{C}^d$. With this definition in place, the general strategy to recover a simple vector $\mathring{\mathbf{x}} \in \mathcal{K} = \operatorname{cone}_k(\mathcal{A})$ from its linear measurements $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}}$ is

$$\begin{aligned} \underset{\mathbf{x}}{\text{minimize}} \quad & \|\mathbf{x}\|_{\mathcal{A}} \\ \text{s.t.} \quad & \mathbf{y} = \mathbf{A}\mathbf{x}. \end{aligned} \tag{P$_{\mathcal{A}}$}$$

We will discuss in Sect. 4 why Problem (P$_{\mathcal{A}}$), which we will simply refer to as *atomic norm minimization*, allows for the recovery of simple sets from underdetermined linear measurements.

In the remainder of this section, we will introduce some of the most common low-complexity sets discussed in the literature. We limit our discussion to sparse vectors, block- and group-sparse vectors, as well as low-rank matrices. Note, however, that the atomic norm framework allows for modeling many other interesting signal classes beyond the ones discussed here. These include permutation and cut matrices, eigenvalue-constrained matrices, low-rank tensors, and binary vectors. We specifically refer interested readers to [31, Sect. 2.2] for a more comprehensive list of example applications of atomic sets.

3.1 Sparse Vectors

As we highlighted various times at this point, the most widespread notion of low complexity at the heart of CS is the notion of sparsity. Even before the advent of compressed sensing, exploiting low complexities in signals played a key role in the development of most compression technologies such as MP3, JPEG, or H264. Ultimately, all these technologies are based on the idea that most signals of interest usually live in rather low-dimensional subspaces embedded in high-dimensional vector spaces.[4] Two canonical examples of this phenomenon are the superposition

[4] This idea also extends to signals living on low-dimensional manifolds.

of sine waves and natural images. In the former case, it is obvious that we are only able to infer very little information from glancing at a time series plot of a sound wave recorded at a microphone. For instance, we might be able to say when a signal is made up of mostly low-frequency components if its waveform only appears to change very slowly over time, but for most signals we are usually not able to say much beyond that. The situation changes drastically, however, if we instead inspect the signal's Fourier transform. In the example of superimposed sine waves, the inherent simplicity or low complexity of the signal becomes immediately apparent in the form of a few isolated peaks in the Fourier spectrum of the signal, revealing the true low-complexity structure of the signal. A similar observation can be made for natural images where periodic structures—say a picture of a garden fence or a brick wall—or flat, homogeneous textures—say in images featuring a view of the sky or blank walls—lead to sparse representations in a variety of bases such as the discrete Fourier transform (DFT) basis, the discrete cosine transform (DCT) basis or the extended family of x-let systems, e.g., wavelets [68], curvelets [22], noiselets [34], shearlets [65], and so on.

Formally, the set of sparse vectors is simply defined as the set of vectors in $\mathbb{C}^d$ with at most k nonzero coefficients. For convenience, this is mostly defined mathematically with the help of the ℓ_0-pseudonorm

$$\|\mathbf{x}\|_0 := |\mathrm{supp}(\mathbf{x})| = |\{i \in [d] : x_i \neq 0\}| \,.$$

With this definition, the set of all k-sparse vectors can be written as

$$\Sigma_k = \{\mathbf{x} : \|\mathbf{x}\|_0 \leq k\}.$$

As we discussed in the beginning of Sect. 3, the set Σ_k is a collection of $\binom{d}{k}$ k-dimensional subspaces, each one spanned by k canonical basis vectors. Since it is a union and not a sum of subspaces, the set is highly nonlinear in nature, e.g., the sum of two k-sparse vectors is generally $2k$-sparse in case the vectors are supported on disjoint support sets.

Consider again the linear inverse problem in which we are tasked with inferring $\mathring{\mathbf{x}} \in \Sigma_k$ from its measurements $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}}$. As we motivated before, if the support of the k-sparse vector is known, so is the corresponding subspace, and the signal can be easily recovered via a least-squares projection. If on the other hand we assume that the support is not known, the situation becomes dire as we now have to consider intractably many possible subspaces. To get a feeling for the complexity of the set of sparse vectors, consider for some $c \in \mathbb{R}$ the set $\left\{\mathbf{x} \in \mathbb{R}^d : \|\mathbf{x}\|_0 = k, x_i = c \;\forall i \in \mathrm{supp}(\mathbf{x})\right\} \subset \Sigma_k$, i.e., the set of exactly k-sparse vectors with identical nonzero entries. A random vector uniformly drawn from this set has entropy $\log\binom{d}{k}$, which means that[5] $\log\binom{d}{k} \approx k \log\,(d/k)$ bits are required for effective compression of this set [90]. As we will see in Sects. 4 and 7, the expression $k\log(d/k)$ plays a key role in the theory of compressed sensing.

[5] This follows from the classical bound $\left(\frac{d}{k}\right)^k \leq \binom{d}{k} \leq \left(\frac{ed}{k}\right)^k$.

To frame the set of sparse vectors in the language of simple sets as established in the beginning of Sect. 3, we note that the atomic set corresponding to the set of sparse vectors in $\mathbb{R}^d$ is simply the set of signed unit vectors, i.e., $\mathcal{A} = \{\pm \mathbf{e}_i\}$.[6] Since the convex hull of $\mathcal{A}$ clearly corresponds to the ℓ_1-unit ball, we have $\Sigma_k(\mathbb{R}^d) = \mathrm{cone}_k(\mathcal{A})$. The atomic norm associated with this set is simply the ℓ_1-norm on $\mathbb{R}^d$. This easily follows from expanding a vector in terms of the elements of $\mathcal{A}$ as

$$\mathbf{x} = \sum_{i=1}^{d} |x_i| \underbrace{\mathrm{sign}(x_i)\mathbf{e}_i}_{\in \mathcal{A}} .$$

Then we have with Definition 5 that

$$\begin{aligned}\|\mathbf{x}\|_{\mathcal{A}} &= \inf \left\{ \sum_{\mathbf{a} \in \mathcal{A}} c_{\mathbf{a}} : \sum_{\mathbf{a} \in \mathcal{A}} c_{\mathbf{a}} \mathbf{a}, c_{\mathbf{a}} \geq 0 \right\} \\ &= \sum_{i=1}^{d} |x_i| = \|\mathbf{x}\|_1 .\end{aligned}$$

While there are infinitely many ways to express each coordinate x_i in terms of nonnegative linear combinations of the atoms $\mathbf{e}_i$ and $-\mathbf{e}_i$, the infimum in the definition of $\|\cdot\|_{\mathcal{A}}$ is attained when each coordinate is expressed by exactly one element of $\mathcal{A}$. This follows immediately from the triangle inequality.

Compressible Vectors

While the concept of sparsity arises naturally in an abundance of contexts and applications, in many cases it is also a slightly too stringent model for practical purposes. A canonical example is natural images which certainly exhibit a low-complexity structure if expressed in a suitable sparsity basis. However, this basis expansion is usually not perfect. In other words, by close inspection one usually notices that while the majority of the signal energy concentrates in only a limited number of expansion coefficients, there usually also exist many coefficients with non-negligible amplitudes which carry information about fine structures of images. Nevertheless, a histogram of the transform coefficients usually reveals that the negligible coefficients quickly decay such that natural images are still be well approximated by sparse vectors. This concept, which leads us to the class of so-called *compressible vectors*, is also heavily exploited in image compression algorithms which quantize infrequently occurring transform coefficients more aggressively (i.e., more coarsely) than more dominant ones such as DC coefficients.

Formally, let $\mathbf{x} \in \mathbb{C}^d$ be a vector whose k largest components in absolute value are supported on a set $S \subset [d]$ of size k, and define for $p > 0$ the *best k-term approximation error* $\sigma_k(\cdot)_p \colon \mathbb{C}^d \to \mathbb{R}_{\geq 0}$ as

[6] To define the sparse vectors on $\mathbb{C}^d$, simply replace $\{\pm \mathbf{e}_n\}$ by $\{\pm \mathbf{e}_n, \pm i \mathbf{e}_n\}$.

$$\sigma_k(\mathbf{x})_p := \min_{\mathbf{z} \in \Sigma_k} \|\mathbf{x} - \mathbf{z}\|_p . \tag{5}$$

For any $p > 0$, the minimum in Eq. (5) is attained by the vector $\mathbf{z}$ which agrees with $\mathbf{x}$ on S and vanishes identically on $\overline{S}$. The following result characterizes the decay behavior of the approximation error.

Theorem 1 ([54, Theorem 2.5]) *Let $q > p > 0$. Then for any $\mathbf{x} \in \mathbb{C}^d$, the best k-term approximation error w. r. t. the ℓ_q-norm is bounded by*

$$\sigma_k(\mathbf{x})_q \leq \frac{c_{p,q}}{k^{1/p-1/q}} \|\mathbf{x}\|_p \tag{6}$$

with

$$c_{p,q} := \exp\left(-\frac{h_b(p/q)}{p}\right) \leq 1,$$

and $h_b(x) := -x\log(x) - (1-x)\log(1-x)$ denoting the binary entropy function. In particular, we have

$$\sigma_k(\mathbf{x})_2 \leq \frac{1}{2\sqrt{k}} \|\mathbf{x}\|_1 .$$

The set of vectors which can be well approximated in terms of σ_k are called *compressible vectors*. Informally, this means that a vector $\mathbf{x}$ is compressible if $\sigma_k(\mathbf{x})_p$ decays quickly as k increases. One particular set of vectors which exhibit such a rapid error decay is the elements of the ℓ_q-quasinorm balls

$$\mathbb{B}_q^d = \left\{\mathbf{z} \in \mathbb{C}^d : \|\mathbf{z}\|_q \leq 1\right\}$$

with $0 < q \leq 1$. To see why the ℓ_q-quasinorm balls are suitable proxies for sparse vectors, consider the limiting behavior of the quasinorm. For $q \to 0$ we have

$$\begin{aligned}
\lim_{q\to 0} \|\mathbf{x}\|_q^q &= \lim_{q\to 0} \sum_{i=1}^{d} |x_i|^q \\
&= \sum_{i=1}^{d} \mathbb{1}_{\{x_i \neq 0\}} \\
&= |\{i \in [d] : x_i \neq 0\}| \\
&= \|\mathbf{x}\|_0 .
\end{aligned}$$

In the other limiting case, one obtains the set of unit ℓ_1-norm vectors. Moreover, applying Theorem 1 to the case of ℓ_q-norm balls, we find

$$\sigma_k(\mathbf{x})_2 \leq \frac{c_{q,2}}{k^{\frac{1}{q}-\frac{1}{2}}}.$$

Finally, it can be shown that the ith biggest entry of $\mathbf{x}$ decays as $i^{-1/q}$ [37].

3.2 Block- and Group-Sparse Vectors

While the model of sparse and compressible vectors has many interesting and justified applications, many times real-world signals will exhibit even more structure beyond simple sparsity. One of the most common generalizations of sparse vectors is so-called *block-sparse* or more generally *group-sparse* signals. In the former case, we assume that the set $[d]$ is partitioned into L disjoint subsets $B_l \subset [d]$ of possibly different sizes $|B_l| = b_l$ such that $\bigcup_{l=1}^{L} B_l = [d]$, and $\sum_{l=1}^{L} b_l = d$. If the sets B_l are allowed to overlap, we refer to them as *groups* instead. As in the case of sparse vectors, a vector $\mathbf{x} \in \mathbb{C}^d$ is called k-block-sparse or k-group-sparse if its nonzero coefficients are limited to at most k nonzero blocks or groups, respectively. Another closely related cousin of block-sparsity is that of fusion frame sparsity. Assuming equisized blocks B_l with $b_l = b$, one additionally imposes in this model that each subvector $\mathbf{x}_{B_l} \in \mathbb{C}^b$ belongs to some s-dimensional subspace $W_l \subset \mathbb{C}^b$ (see, e.g., [5, 15], for details). Structured sparsity models as outlined above arise in a variety of domains in engineering and biology. Some prominent example applications are audio [1] and image signal processing [102], multi-band reconstruction and spectrum sensing [70, 81], as well as sparse subspace clustering [48]. Further applications in which block- and group-sparse signal structures commonly appear are in the context of measuring gene expression levels [78] and protein mass spectroscopy [93]. For a more thorough treatment of block-sparse signal modeling, we also refer readers to [47, Chap. 2].

In the following, we limit our discussion to the case of block-sparsity. A natural way to express the block-sparsity of a vector mathematically is by introducing for $p, q > 0$ the family of mixed (ℓ_p, ℓ_q)-(quasi)norms

$$\|\mathbf{x}\|_{p,q} := \left(\sum_{l=1}^{L} \left\| \mathbf{x}_{B_l} \right\|_p^q \right)^{1/q},$$

where we denote by $\mathbf{x}_{B_l} \in \mathbb{C}^d$ the subvector of $\mathbf{x}$ restricted to the index set B_l. Extending the notation to include the case $q = 0$, we define additionally the mixed (ℓ_p, ℓ_0)-pseudonorm

$$\begin{aligned} \|\mathbf{x}\|_{p,0} &:= \left| \left\{ \left\| \mathbf{x}_{B_l} \right\|_p \neq 0 : l \in [L] \right\} \right| \\ &= \left| \left\{ \mathbf{x}_{B_l} \neq \mathbf{0} : l \in [L] \right\} \right|, \end{aligned}$$

which simply counts the number of nonzero blocks of $\mathbf{x}$ w.r.t. $\{B_l\}_{l=1}^L$. With this definition, a vector is called k-block-sparse if $\|\mathbf{x}\|_{p,0} \le k$. Moreover, the atomic set which gives rise to the set of k-block-sparse vectors can now be defined as

$$\mathcal{A}_p := \bigcup_{l=1}^{L} \left\{ \mathbf{a} \in \mathbb{C}^d : \left\|\mathbf{a}_{B_l}\right\|_p = 1, \mathbf{a}_{\overline{B_l}} = \mathbf{0} \right\}. \tag{7}$$

Note that unlike in the case of sparse vectors where we defined $\tilde{\mathcal{A}} = \{\pm \mathbf{e}_i\}$, the set in Eq. (7) is uncountable. To calculate the atomic norm, recall the definition

$$\|\mathbf{x}\|_{\mathcal{A}_p} = \inf \left\{ \sum_{\mathbf{a}\in\mathcal{A}} c_{\mathbf{a}} : \mathbf{x} = \sum_{\mathbf{a}\in\mathcal{A}} c_{\mathbf{a}}\mathbf{a},\, c_{\mathbf{a}} \ge 0 \right\}.$$

Since $\operatorname{span}(\mathcal{A}_p) = \mathbb{C}^d$, there exists a $c_{\mathbf{a}} \ge 0$ and $\mathbf{a} \in \mathcal{A}_p$ such that for every $\mathbf{x} \in \mathbb{C}^d$, we may express its coefficients in block B_l as $\mathbf{x}_{B_l} = c_{\mathbf{a}}\mathbf{a}$. Then we have $\left\|\mathbf{x}_{B_l}\right\|_p = \|c_{\mathbf{a}}\mathbf{a}\|_p = |c_{\mathbf{a}}| \cdot \|\mathbf{a}\|_p = c_{\mathbf{a}}$ where the last step simply follows from the fact that $c_{\mathbf{a}} \ge 0$ and $\mathbf{a} \in \mathcal{A}_p$. Again, we have by the triangle inequality that the infimum in the definition of the atomic norm must be attained by a decomposition where each block B_l is represented by exactly one atom. Hence

$$\|\mathbf{x}\|_{\mathcal{A}_p} = \sum_{l=1}^{L} \left\|\mathbf{x}_{B_l}\right\|_p = \|\mathbf{x}\|_{p,1}.$$

Note that a similar argument holds for the group-sparsity case where the sets B_l are not assumed to be disjoint [84, Lemma 2.1].

Clearly, the atomic norm induced by $\mathcal{A}$ is closely related to the ℓ_1-norm as discussed in the previous section. In the edge case with $L = d$, and $|B_l| = 1$, we have $\mathcal{A}_p = \{\pm \mathbf{e}_i\}$ such that we immediately arrive again at the set of sparse vectors.

3.3 Low-Rank Matrices

A slightly different linear inverse problem which can still be conveniently modeled by means of atomic sets is the so-called low-rank matrix recovery problem. Consider a matrix $\mathbf{X} \in \mathbb{C}^{d_1 \times d_2}$ of rank at most r which we observe through the linear operator

$$\mathcal{M}\colon \mathbb{C}^{d_1\times d_2} \to \mathbb{C}^m : \mathbf{X} \mapsto \mathcal{M}(\mathbf{X}) = \mathbf{y}.$$

As usual, our task is to infer $\mathbf{X}$ from knowledge of the map $\mathcal{M}$ and the measurements $\mathbf{y}$ by solving the atomic norm minimization problem ($\mathrm{P}_{\mathcal{A}}$). In general, there are of course $d_1 d_2$ unknown entries in $\mathbf{X}$ so that the linear inverse problem is clearly ill-posed as long as $m < d_1 d_2$. However, by exploiting a potential low-rank structure on $\mathbf{X}$, it turns out to be possible to drastically reduce the number of observations needed to allow for faithful estimation of low-rank matrices (cf. Table 1).

A typical example application of low-rank matrix recovery, known as the *matrix completion* problem, is the task of estimating missing entries of a matrix based on partial observations of $\mathbf{X}$ of the form $\mathcal{M}(\mathbf{X})_i = X_{kl}$ for some $(k, l) \in [d_1] \times [d_2]$. As before, this problem is clearly hopelessly ill-posed if $\mathbf{X}$ is a full-rank or close to full-rank matrix. However, in many practical situations in the context of collaborative filtering [56], the low-rank assumption on $\mathbf{X}$ is justified by the problem domain, making low-rank matrix recovery a useful prediction tool. The matrix completion problem was famously popularized by the so-called *Netflix Prize* [11], an open competition in collaborative filtering to predict user ratings of movies based on partial knowledge of ratings about other titles in the portfolio. The underlying assumption is that if two users both share the same opinion about certain titles they saw, then they are likely to share the same opinion about titles so far only seen or rated by one of them. In other words, if we collect the user ratings of all available titles in a database in a matrix $\mathbf{X}$, then we can assume that due to overlapping interests and opinions, the matrix will exhibit a low-rank structure. This reduction in the degrees of freedom therefore allows to accurately predict unknown user ratings which can then be used to provide personalized recommendations on a per-user basis.

To demonstrate how low-rank matrices can be modeled in the context of atomic sets, consider the set of rank-1 matrices of the form

$$\begin{aligned}\mathcal{A} &= \left\{\mathbf{u}\mathbf{v}^* \in \mathbb{C}^{d_1 \times d_2} : \|\mathbf{u}\|_2 = \|\mathbf{v}\|_2 = 1\right\} \\ &= \left\{\mathbf{u}\mathbf{v}^* \in \mathbb{C}^{d_1 \times d_2} : \left\|\mathbf{u}\mathbf{v}^*\right\|_{\mathrm{F}} = 1\right\}.\end{aligned}$$

Clearly, a nonnegative linear combination of r elements of $\mathcal{A}$ forms a matrix of at most rank r so that $\operatorname{cone}_r(\mathcal{A})$ generates the set of rank r matrices. To derive the atomic norm associated with $\mathcal{A}$, consider that for every $\mathbf{X} \in \mathbb{C}^{d_1 \times d_2}$ we have by the singular value decomposition of $\mathbf{X}$ that

$$\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^*,$$

Table 1 Mean width estimates for tangent cones

Signal set	Induced norm	Upper bound on $w(\mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}}) \cap \mathbb{S}^{d-1})^2$
Sparse vectors in $\mathbb{R}^d$	$\|\cdot\|_1$	$2k \log(d/k) + 3k/2$
Block-sparse vectors in $\mathbb{R}^d$ with L blocks of size d/L	$\|\cdot\|_{2,1}$	$4k \log(L/k) + (1 + 6d/L)k/2$
Rank r matrices in $\mathbb{R}^{d_1 \times d_2}$	$\|\cdot\|_*$	$3r(d_1 + d_2 - r)$

where $\mathbf{U} \in \mathbb{C}^{d_1 \times d_1}$ and $\mathbf{V} \in \mathbb{C}^{d_2 \times d_2}$ are unitary matrices, and $\Sigma \in \mathbb{C}^{d_1 \times d_2}$ is a matrix containing the real-valued, nonnegative singular values on its main diagonal and zeros otherwise. Hence, we have with $d := \min\{d_1, d_2\}$,

$$\mathbf{X} = \sum_{i=1}^{d} \sigma_i \mathbf{u}_i \mathbf{v}_i^*$$

with $\mathbf{u}_i \mathbf{v}_i^* \in \mathcal{A}$. Again, with Definition 5 this yields

$$\begin{aligned} \|\mathbf{X}\|_{\mathcal{A}} &= \inf \left\{ \sum_{\mathbf{a} \in \mathcal{A}} c_{\mathbf{a}} : \mathbf{X} = \sum_{\mathbf{a} \in \mathcal{A}} c_{\mathbf{a}} \mathbf{a}, \, c_{\mathbf{a}} \geq 0 \right\} \\ &= \sum_{i=1}^{d} \sigma_i(\mathbf{X}) =: \|\mathbf{X}\|_*, \end{aligned}$$

where in the second step we simply identified $c_{\mathbf{a}}$ with the singular values of the decomposition after using the fact that by the triangle inequality (w. r. t. the Frobenius norm), the infimum must be attained by a decomposition of at most d atoms. While the singular vectors $\mathbf{u}_i$ and $\mathbf{v}_i$ which make up the atoms $\mathbf{a} = \mathbf{u}_i \mathbf{v}_i^* \in \mathcal{A}$ are not necessarily unique, each $\mathbf{X}$ is identified by a unique set of singular values.

The norm $\|\cdot\|_*$ is generally known as the nuclear norm and acts as an analog of the ℓ_1-norm in the case of sparse vectors since $\|\mathbf{X}\|_*$ corresponds to the ℓ_1-norm of the vector of singular values of $\mathbf{X}$. Considering that efficient algorithms for the singular value decomposition exist, the atomic norm minimization for low-rank matrices constitutes a tractable convex optimization problem.

Representability of Atomic Norms

While the examples of atomic sets we presented so far all admitted relatively straightforward representations of their associated atomic norms, efficient computation of $\|\cdot\|_{\mathcal{A}}$ for arbitrary atomic sets $\mathcal{A}$ is by no means guaranteed. A classic example of where the atomic norm framework fails to yield an efficient way to recover elements of a simple set generated by $\text{cone}_k(\mathcal{A})$ is the set

$$\mathcal{A} = \left\{ \mathbf{z}\mathbf{z}^\top : \mathbf{z} \in \{\pm 1\}^d \right\}.$$

Similar to the set of low-rank matrices, the simple set generated by $\mathcal{A}$ consists of low-rank matrices but with its elements restricted to the set ± 1—a model which appears, for instance, in the context of collaborative filtering [73]. Considering that $\text{conv}(\mathcal{A})$ corresponds to the so-called *cut polytope* which does not admit a tractable characterization, there exists no efficient way of computing $\|\cdot\|_{\mathcal{A}}$. In this case, one may turn to a particular approximation scheme of $\text{conv}(\mathcal{A})$ known as *theta bodies* [58] which are closely related to the theory of sum-of-squares (SOS) polynomials. We refer interested readers to [31, Sect. 4].

As another example, consider the atomic set

$$\mathcal{A} = \left\{\mathbf{a}_{f,\phi} \in \mathbb{C}^d : f \in [0, 1], \phi \in [0, 2\pi)\right\}$$

with

$$\mathbf{a}_{f,\phi} := e^{i2\pi\phi} \begin{pmatrix} 1 \\ e^{i2\pi f} \\ \vdots \\ e^{i2\pi f(d-1)} \end{pmatrix}.$$

This set represents a continuous alphabet of atoms which gives rise to the signal set of sampled representations of continuous-time superpositions of complex exponentials [13]. Using results from the theory of SOS polynomials, Bhaskar et al. showed in [13] that the associated atomic norm can be computed as the solution of the program

$$\begin{array}{ll} \underset{\mathbf{x},\mathbf{u},t}{\text{minimize}} & \dfrac{\operatorname{tr} T(\mathbf{u})}{2d} + \dfrac{t}{2} \\ \text{s.t.} & \begin{pmatrix} T(\mathbf{u}) & \mathbf{x} \\ \mathbf{x}^* & t \end{pmatrix} \succeq 0 \end{array}$$

where the linear operator $T: \mathbb{C}^d \to \mathbb{C}^{d \times d}$ maps a vector $\mathbf{u}$ to the Toeplitz matrix generated by $\mathbf{u}$. The same representation also appears in the context of *compressed sensing off the grid* where one aims to recover a sampled representation of a superposition of complex exponentials from randomly observed time-domain samples [92].

Both of these examples illustrate that while the atomic norm framework represents a convenient modeling tool for low-complexity signal sets, it may turn out to be a nontrivial or in some cases simply impossible task to actually find efficient ways to compute the atomic norm.

3.4 Low-Complexity Models in Bases and Frames

Up until this point, we have assumed that signals of interest are elements of a simple set $\mathcal{K} = \operatorname{cone}_k(\mathcal{A})$ generated by an atomic set $\mathcal{A}$. Given a vector $\mathring{\mathbf{x}} \in \mathcal{K}$ and its linear measurements $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}}$, the general task is to infer $\mathring{\mathbf{x}}$ from knowledge of $\mathbf{A}$ and $\mathbf{y}$. In this context, the measurement process is entirely modeled by $\mathbf{A}$. However, oftentimes in practical scenarios, we might not have direct access to the signal exhibiting a low-complexity structure but rather only to its representation in a particular *orthonormal basis* or more generally an *overcomplete dictionary* or *frame*. As a classical example, consider the situation in which $\mathring{\mathbf{x}} \in \mathbb{C}^d$ represents the sampled time-domain representation of a band-limited function. If the continuous-time signal is a superposition of k complex exponentials, the sampled representation $\mathring{\mathbf{x}}$

will generally have dense support. The underlying sparsity structure[7] only reveals itself to us after transforming $\mathring{\mathbf{x}}$ into the frequency domain, i.e., $\mathring{\mathbf{z}} = \mathbf{F}_d\mathring{\mathbf{x}} \in \Sigma_k$ with $\mathbf{F}_d = d^{-1/2}(e^{-i2\pi\mu\nu})_{0\leq\mu,\nu\leq d-1}$ denoting the DFT matrix. We therefore acquire measurements according to $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}} = \mathbf{A}\mathbf{F}_d^*\mathring{\mathbf{z}} =: \tilde{\mathbf{A}}\mathring{\mathbf{z}}$. Reconstruction of $\mathring{\mathbf{x}}$ now proceeds in two steps by first reconstructing the vector $\mathring{\mathbf{z}}$, exploiting its underlying low-complexity structure, and then resynthesizing the estimate of $\mathring{\mathbf{x}}$. For this reason, this model is also known as *synthesis model* throughout the literature. In general, one may assume that rather than exhibiting a low-complexity structure in the canonical basis, applications typically either fix or learn a suitable basis change matrix. Moreover, allowing for the transform matrix to be an overcomplete dictionary or frame $\mathbf{\Omega} \in \mathbb{C}^{d\times D}$ with $D > d$ such that $\mathring{\mathbf{x}} = \mathbf{\Omega}\mathring{\mathbf{z}}$ where $\mathring{\mathbf{z}} \in \mathbb{C}^D$ exhibits a low-complexity structure, one may exploit additional advantages stemming from the redundancy of overcomplete representations [30]. Classical examples of such representation systems are curvelet transforms [23] and time–frequency atoms arising from the Gabor transform [49]. For simplicity of presentation, we will assume in the remainder of this chapter that signals of interest already live in simple sets, i.e., we set $\mathbf{\Omega} = \mathbf{I}_d$, and point out that most results presented in the sequel also generalize to low-complexity models in unitary bases and frames. For more details, we refer interested readers to [86].

4 Recovery of Individual Vectors

In this section, we address the recovery of individual signals in simple sets $\mathcal{K}$ generated by $\text{cone}_k(\mathcal{A})$. For simplicity, we limit our discussion to the case where the atomic set $\mathcal{A}$ contains only real elements so that $\mathcal{K} \subset \mathbb{R}^d$.

4.1 Exact Recovery

We begin our discussion by motivating why atomic norm minimization as stated in Problem ($\mathrm{P}_{\mathcal{A}}$) is a suitable strategy for the recovery of simple signals from linear measurements. To that end, consider again the equality-constrained minimization problem

$$\begin{aligned} &\text{minimize } \|\mathbf{x}\|_{\mathcal{A}} \\ &\text{s.t.} \qquad \mathbf{A}\mathring{\mathbf{x}} = \mathbf{A}\mathbf{x}. \end{aligned} \tag{8}$$

By rewriting the equality constraint in terms of $\mathbf{d} = \mathring{\mathbf{x}} - \mathbf{x} \in \ker(\mathbf{A})$, we may restate the problem as

[7] We assume that the fundamental frequencies of each complex exponential are integer multiples of the frequency resolution f_s/d where f_s denotes the sampling rate.

$$\underset{\mathbf{d} \in \ker(\mathbf{A})}{\text{minimize}} \quad \left\| \mathbf{d} + \mathring{\mathbf{x}} \right\|_{\mathcal{A}} .$$

Of course, the above problem is not of any practical interest as it requires knowledge of the true solution $\mathring{\mathbf{x}}$. However, it immediately follows from this representation that Problem (8) has a unique solution if the null space of $\mathbf{A}$ does not contain any nontrivial directions which reduce the atomic norm anchored at $\mathring{\mathbf{x}}$. More precisely, by introducing the set

$$\mathcal{D}_{\mathcal{A}}(\mathring{\mathbf{x}}) := \left\{ \mathbf{d} \in \mathbb{R}^d : \left\| \mathbf{d} + \mathring{\mathbf{x}} \right\|_{\mathcal{A}} \leq \left\| \mathring{\mathbf{x}} \right\|_{\mathcal{A}} \right\} = \left\{ \mathbf{z} - \mathring{\mathbf{x}} : \|\mathbf{z}\|_{\mathcal{A}} \leq \left\| \mathring{\mathbf{x}} \right\|_{\mathcal{A}} \right\}$$

of *descent directions* of $\|\cdot\|_{\mathcal{A}}$ at $\mathring{\mathbf{x}}$, we obtain the condition

$$\mathcal{D}_{\mathcal{A}}(\mathring{\mathbf{x}}) \cap \ker(\mathbf{A}) = \{\mathbf{0}\}, \tag{9}$$

which, if satisfied, guarantees perfect recovery of $\mathring{\mathbf{x}}$ via Problem (8).

Alternatively, one may argue as follows. Let $\mathring{\mathbf{x}} \in \operatorname{cone}_k(\mathcal{A})$ and define the set $\mathcal{X} = \left\| \mathring{\mathbf{x}} \right\|_{\mathcal{A}} \operatorname{conv}(\mathcal{A})$ which clearly contains $\mathring{\mathbf{x}}$. Given access to linear measurements of the form $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}}$, one may then attempt to solve the feasibility problem

$$\begin{aligned} &\text{find } \mathbf{x} \in \mathcal{X} \\ &\text{s.t. } \ \mathbf{y} = \mathbf{A}\mathbf{x} \end{aligned} \tag{10}$$

to recover $\mathring{\mathbf{x}}$. This program has a unique solution if $\mathcal{X}$ intersects the affine subspace $E_{\mathring{\mathbf{x}}} := \left\{ \mathbf{z} \in \mathbb{R}^d : \mathbf{A}\mathbf{z} = \mathbf{A}\mathring{\mathbf{x}} \right\}$ only at the solution $\mathring{\mathbf{x}}$, i.e.,

$$\begin{aligned} \mathcal{X} \cap E_{\mathring{\mathbf{x}}} &= \left\{ \mathring{\mathbf{x}} \right\} \\ \iff (\mathcal{X} - \mathring{\mathbf{x}}) \cap (E_{\mathring{\mathbf{x}}} - \mathring{\mathbf{x}}) &= \{\mathbf{0}\} \\ \iff (\mathcal{X} - \mathring{\mathbf{x}}) \cap \ker(\mathbf{A}) &= \{\mathbf{0}\}. \end{aligned} \tag{11}$$

Since Definition 4 required $\operatorname{conv}(\mathcal{A})$ to be a symmetric convex body, it is also a closed star domain.[8] In this case, we may use a well-known result from functional analysis that allows us to express $\mathcal{X}$ in terms of the 1-sublevel set of its Minkowski functional [88]

$$\begin{aligned} \gamma_{\mathcal{X}}(\mathbf{x}) &= \inf \left\{ t > 0 : \mathbf{x} \in t \left\| \mathring{\mathbf{x}} \right\|_{\mathcal{A}} \operatorname{conv}(\mathcal{A}) \right\} \\ &= \frac{1}{\left\| \mathring{\mathbf{x}} \right\|_{\mathcal{A}}} \inf \left\{ t > 0 : \mathbf{x} \in t \operatorname{conv}(\mathcal{A}) \right\} = \frac{\|\mathbf{x}\|_{\mathcal{A}}}{\left\| \mathring{\mathbf{x}} \right\|_{\mathcal{A}}}. \end{aligned}$$

Thus we have that

[8] A set K is a closed star domain if K is closed, and $tK \subseteq K \ \forall t \in [0, 1]$.

$$\begin{aligned}\mathcal{X} - \mathring{\mathbf{x}} &= \left\{\mathbf{x} \in \mathbb{R}^d : \gamma_{\mathcal{X}}(\mathbf{x}) \leq 1\right\} - \mathring{\mathbf{x}} \\ &= \left\{\mathbf{x} - \mathring{\mathbf{x}} : \|\mathbf{x}\|_{\mathcal{A}} \leq \left\|\mathring{\mathbf{x}}\right\|_{\mathcal{A}}\right\} \\ &= \mathcal{D}_{\mathcal{A}}(\mathring{\mathbf{x}}),\end{aligned}$$

yielding again the uniqueness condition stated in Eq. (9).

Since $\|\cdot\|_{\mathcal{A}}$ defines a norm on $\mathbb{R}^d$, the set of descent directions is a convex body. We may therefore replace $\mathcal{D}_{\mathcal{A}}(\mathring{\mathbf{x}})$ in Eq. (9) by its conic hull without changing the statement. This set, denoted by

$$\mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}}) := \operatorname{cone} \mathcal{D}_{\mathcal{A}}(\mathring{\mathbf{x}}),$$

is usually referred to as the *tangent* or *descent cone* of $\|\cdot\|_{\mathcal{A}}$ at $\mathring{\mathbf{x}}$, and represents a central object in the study of convex analysis. This ultimately leads to the following result.

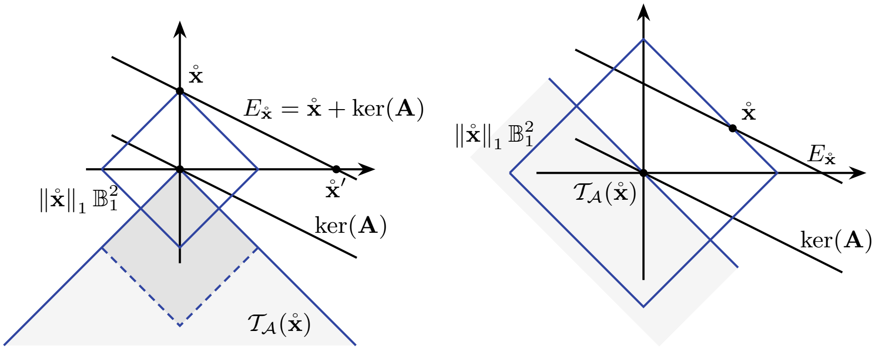
Proposition 1 ([13, Proposition 2.1]) *The vector* $\mathring{\mathbf{x}}$ *is the unique solution of Problem* ($\mathrm{P}_{\mathcal{A}}$) *if and only if*

$$\mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}}) \cap \ker(\mathbf{A}) = \{\mathbf{0}\}. \tag{12}$$

As a typical example application of Proposition 1, consider the atomic set $\mathcal{A} = \{\pm\mathbf{e}_i\} \subset \mathbb{R}^d$ of signed unit vectors. The convex hull of this set is the ℓ_1-unit ball in $\mathbb{R}^d$, and hence $\|\cdot\|_{\mathcal{A}} = \|\cdot\|_1$; the conic hull is all of $\mathbb{R}^d$. However, if we restrict attention to nonnegative linear combinations of at most k elements in $\mathcal{A}$, we obtain the set $\mathcal{K} = \operatorname{cone}_k(\mathcal{A}) = \left\{\mathbf{x} \in \mathbb{R}^d : |\operatorname{supp}(\mathbf{x})| \leq k\right\} = \Sigma_k(\mathbb{R}^d)$ of k-sparse vectors. As illustrated in Fig. 2a, the 1-sparse vector $\mathring{\mathbf{x}}$ can be uniquely recovered via ℓ_1-minimization since its tangent cone $\mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}})$ intersects the null space of $\mathbf{A}$ only at $\{\mathbf{0}\}$. On the other hand, if $\mathring{\mathbf{x}}$ is as depicted in Fig. 2b, then the tangent cone of $\mathcal{A}$ at $\mathring{\mathbf{x}}$ corresponds to a rotated half-space. Since every 1-dimensional subspace of $\mathbb{R}^2$ clearly intersects this half-space at arbitrarily many points, the only way a vector on a 2-dimensional face of $\left\|\mathring{\mathbf{x}}\right\|_1 \mathbb{B}_1^2$ can be recovered is if $\ker(\mathbf{A})$ is the 0-dimensional subspace $\{\mathbf{0}\}$, i.e., if $\mathbf{A}$ has full-rank. Finally, note that the vector $\mathring{\mathbf{x}}'$ in Fig. 2a cannot be recovered either despite sharing the same sparsity structure as $\mathring{\mathbf{x}}$. Conceptually, this is immediately obvious from the fact that $\|\mathring{\mathbf{x}}\|_1 < \|\mathring{\mathbf{x}}'\|_1$ which implies that even if we were to observe $\mathring{\mathbf{x}}'$, atomic norm minimization would still yield the solution $\mathbf{x}^\star = \mathring{\mathbf{x}}$. In light of Proposition 1, this is explained by the fact that the tangent cone at $\mathring{\mathbf{x}}'$ has the same shape as $\mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}})$ but rotated 90° clockwise so that $\mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}}')$ and $\ker(\mathbf{A})$ share a ray, violating the uniqueness condition (12). This example demonstrates the nonuniform character of the recovery condition of Proposition 1 which locally depends on the particular choice of $\mathring{\mathbf{x}}$.

Since the tangent cone is a bigger set than $\mathcal{D}_{\mathcal{A}}(\mathring{\mathbf{x}})$, the condition

$$\mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}}) \cap \ker(\mathbf{A}) = \{\mathbf{0}\}$$


$\mathring{\mathbf{x}}$
$E_{\mathring{\mathbf{x}}} = \mathring{\mathbf{x}} + \ker(\mathbf{A})$
$\mathring{\mathbf{x}}'$
$\|\mathring{\mathbf{x}}\|_1\, \mathbb{B}_1^2$
$\ker(\mathbf{A})$
$\mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}})$
$\|\mathring{\mathbf{x}}\|_1\, \mathbb{B}_1^2$
$\mathring{\mathbf{x}}$
$E_{\mathring{\mathbf{x}}}$
$\mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}})$
$\ker(\mathbf{A})$

Recovery of 1-sparse vectors

Recovery of a 2-sparse vector

Fig. 2 Recovery of vectors in $\mathbb{R}^2$

in a sense represents a stronger requirement than $\mathcal{D}_{\mathcal{A}}(\mathring{\mathbf{x}}) \cap \ker(\mathbf{A})$ from before. Moreover, while Proposition 1 provides a necessary and sufficient condition for the successful recovery of individual vectors via Problem ($\mathrm{P}_{\mathcal{A}}$), testing the condition in practice ultimately requires prior knowledge of the solution $\mathring{\mathbf{x}}$ which we aim to recover. However, as we will see shortly, both issues can be elegantly circumvented by turning to the probabilistic setting where we assume the elements of the measurement matrix are drawn independently from the standard Gaussian distribution. This will allow us to draw on a powerful result from asymptotic convex geometry to assess the success of recovering individual vectors probabilistically. Before stating this result, we first need to introduce the concept of *Gaussian mean width* or *mean width* for short, an important summary parameter of a bounded set.

Definition 6 (*Gaussian mean width*) The Gaussian mean width of a bounded set $\boldsymbol{\Omega}$ is defined as

$$w(\boldsymbol{\Omega}) := \mathbb{E} \sup_{\mathbf{x} \in \boldsymbol{\Omega}} \langle \mathbf{g}, \mathbf{x} \rangle, \tag{13}$$

where $\mathbf{g} \sim \mathsf{N}(\mathbf{0}, \mathrm{Id})$ is an isotropic zero-mean Gaussian random vector.

The Gaussian mean width is closely related to the spherical mean width

$$w_{\mathbb{S}}(\boldsymbol{\Omega}) := \mathbb{E} \sup_{\mathbf{x} \in \boldsymbol{\Omega}} \langle \boldsymbol{\eta}, \mathbf{x} \rangle,$$

where $\boldsymbol{\eta}$ is a random d-vector drawn uniformly from the Haar measure on the sphere. Since length and direction of a Gaussian random vector are independent by rotation invariance of the Gaussian distribution, we can decompose every standard Gaussian vector $\mathbf{g}$ as $\mathbf{g} = \|\mathbf{g}\|_2\, \boldsymbol{\eta}$ where $\boldsymbol{\eta}$ is again drawn from the uniform Haar measure. The Gaussian and spherical mean width are therefore related by

$$w(\boldsymbol{\Omega}) = \mathbb{E}\,\|\mathbf{g}\|_2\, w_{\mathbb{S}}(\boldsymbol{\Omega}) \leq \sqrt{d} w_{\mathbb{S}}(\boldsymbol{\Omega}),$$

where the last step follows from Jensen's inequality. Intuitively, the mean width of a bounded set measures its average diameter over all directions chosen uniformly at random. Consider for a moment the mean width $w(\boldsymbol{\Omega} - \boldsymbol{\Omega})$ of the Minkowski difference of $\boldsymbol{\Omega}$ with itself. Then we immediately have

$$\begin{aligned} w(\boldsymbol{\Omega} - \boldsymbol{\Omega}) &= \mathbb{E} \sup_{\mathbf{d} \in \boldsymbol{\Omega} - \boldsymbol{\Omega}} \langle \mathbf{g}, \mathbf{d} \rangle \\ &= \mathbb{E} \sup_{\mathbf{x}, \mathbf{z} \in \boldsymbol{\Omega}} \langle \mathbf{g}, \mathbf{x} - \mathbf{z} \rangle \\ &\leq 2\mathbb{E} \sup_{\mathbf{x} \in \boldsymbol{\Omega}} \langle \mathbf{g}, \mathbf{x} \rangle = 2w(\boldsymbol{\Omega}) \end{aligned}$$

with equality if $\boldsymbol{\Omega}$ is origin-symmetric. Given a realization of the random vector $\mathbf{g}$, the term $\sup_{\mathbf{x},\mathbf{z} \in \boldsymbol{\Omega}} \langle \mathbf{g}, \mathbf{x} - \mathbf{z} \rangle$ then corresponds to the distance of two supporting hyperplanes to $\boldsymbol{\Omega}$ with normal $\mathbf{g}$, scaled by $\|\mathbf{g}\|_2$.

With the definition of the mean width in place, we are now ready to state the following result known as *Gordon's escape through a mesh* or simply *Gordon's escape theorem*. We present here a version of the theorem adopted from [31, Corollary 3.3]. The original result was first presented in [57].

Theorem 2 (Gordon's escape through a mesh) *Let $S \subset \mathbb{S}^{d-1}$, and let E be a random $(d-m)$-dimensional subspace of $\mathbb{R}^d$ drawn uniformly from the Haar measure on the Grassmann manifold $\mathcal{G}(d, d-m)$.*[9] *Then*

$$\mathbb{P}(S \cap E = \emptyset) \geq 1 - \exp\left(-\frac{1}{2} \left[\frac{m}{\sqrt{m+1}} - w(S) \right]^2 \right)$$

provided

$$m \geq w(S)^2 + 1.$$

In words, Gordon's escape through a mesh phenomenon asserts that a randomly drawn subspace misses a subset of the Euclidean unit sphere with overwhelmingly high probability if the codimension m of the subspace is on the order of $w(S)^2$. Moreover, the probability of this event only depends on the codimension m of the subspace, as well as on the Gaussian width of the sphere patch S. In order to apply this result to the situation of Proposition 1 in the context of the standard Gaussian measurement ensemble, we merely need to restrict the tangent cone $\mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}})$ to the sphere, i.e., $S = \mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}}) \cap \mathbb{S}^{d-1}$, and choose $E = \ker(\mathbf{A})$. This immediately yields the following straightforward specialization of Theorem 2.

[9] The *Grassmann manifold* or *Grassmannian* $\mathcal{G}(d, s)$ is an abstract Riemannian manifold containing all s-dimensional subspaces of $\mathbb{R}^d$.

Corollary 1 (Exact recovery from Gaussian observations) *Let* $\mathbf{A} \in \mathbb{R}^{m\times d}$ *be a matrix populated with independent standard Gaussian entries, and let* $\mathring{\mathbf{x}} \in \operatorname{cone}_k(\mathcal{A})$*. Then* $\mathring{\mathbf{x}}$ *can be perfectly recovered from its measurements* $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}}$ *via atomic norm minimization with probability at least* $1 - \eta$ *if*

$$m \geq \left(w(\mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}}) \cap \mathbb{S}^{d-1}) + \sqrt{2\log(\eta^{-1})}\right)^2.$$

So far, we have only concerned ourselves with establishing conditions under which an arbitrary vector could be uniquely recovered from its linear measurements by solving Problem (8). In fact, nothing in our discussion so far precludes that this undertaking might require us to take at least as many measurements as the linear algebraic dimension of the vector space containing $\mathring{\mathbf{x}}$. The power of the presented approach lies in the fact that for many signal models of interest such as sparse vectors, group-sparse vectors, and low-rank matrices, the tangent cone at points $\mathring{\mathbf{x}}$ lying on low-dimensional faces of a scaled version of $\operatorname{conv}(\mathcal{A})$ is narrow (cf. Fig. 2), and therefore exhibit small mean widths. Coming back to the canonical example of sparse vectors as discussed before, it can be shown that $w(\mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}}) \cap \mathbb{S}^{d-1})$ roughly scales like $\sqrt{k\log(d/k)}$ for any $\mathring{\mathbf{x}} \in \Sigma_k(\mathbb{R}^d)$ (see, for instance, [31, 89]). In light of Corollary 1, this requires m to scale linearly in k, and only logarithmically in the ambient dimension d. For convenience, we list some of the best known bounds for the mean widths of tangent cones associated with the signal models introduced in Sect. 3 in Table 1 [55].

Without going into too much detail, we want to briefly comment on a few natural extensions of Corollary 1.

Extensions to Noisy Recovery and Subgaussian Observations

An obvious question to ask at this point is what kind of recovery performance we might expect if we extend our sensing model to include additive noise of the form $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}} + \mathbf{w}$ with $\|\mathbf{w}\|_2 \leq \sigma$ as a more realistic model of observation. Naturally, we cannot hope to ever recover $\mathring{\mathbf{x}}$ exactly in that case unless $\sigma = 0$. Nevertheless, one should still expect to be able to control the recovery quality in terms of the mean width of the tangent cone and the noise level σ by an appropriate choice of m. The following result, which was adapted from [31, Corollary 3.3], demonstrates that this is in fact the case if we solve the noise-constrained atomic norm minimization problem

$$\begin{aligned} \text{minimize} \quad & \|\mathbf{x}\|_{\mathcal{A}} \\ \text{s.t.} \quad & \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2 \leq \sigma. \end{aligned} \tag{14}$$

Proposition 2 (Robust recovery from Gaussian observations) *Let* $\mathbf{A}$ *and* $\mathring{\mathbf{x}}$ *be as in Corollary 1. Assume we observe* $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}} + \mathbf{w}$ *with* $\|\mathbf{w}\|_2 \leq \sigma$*. Then with probability at least* $1 - \eta$*, the solution* $\mathbf{x}^\star$ *of Problem* (14) *satisfies*

$$\left\| \mathring{\mathbf{x}} - \mathbf{x}^\star \right\|_2 \le \nu$$

provided

$$m \ge \left(\frac{w(\mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}}) \cap \mathbb{S}^{d-1}) + \sqrt{2\log(\eta^{-1})}}{1 - 2\sigma/\nu} \right)^2 .$$

Note that the reconstruction fidelity ν in Proposition 2 is inherently limited by the noise level σ since we require $\nu > 2\sigma$ for the bound on m to yield sensible values.

In closing, we also want to mention a recent extension of Gordon's escape theorem to measurement matrices whose rows are independent copies of subgaussian isotropic random vectors $\mathbf{a}_i \in \mathbb{R}^d$ with subgaussian parameter τ, i.e.,

$$\mathbb{E}(\mathbf{a}_i \mathbf{a}_i^\top) = \mathrm{Id}, \quad \|\mathbf{a}_i\|_{\psi_2} = \sup_{\boldsymbol{\theta} \in \mathbb{S}^{d-1}} \|\langle \boldsymbol{\theta}, \mathbf{a}_i \rangle\|_{\psi_2} \le \tau. \tag{15}$$

Based on a concentration result for such matrices acting on bounded subsets of $\mathbb{R}^d$ [66, Corollary 1.5], Liaw et al. proved a general version of the following result which we state here in the context of signal recovery in the same vein as Corollary 1.

Theorem 3 (Exact recovery from subgaussian observations) *Let* $\mathbf{A} \in \mathbb{R}^{m \times d}$ *be a matrix whose rows are independent subgaussian random vectors satisfying Eq.* (15)*, and let* $\mathring{\mathbf{x}} \in \mathrm{cone}_k(\mathcal{A})$*. Then with probability at least* $1 - \eta$*,* $\mathring{\mathbf{x}}$ *is the unique minimizer of Problem* (8) *with* $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}}$ *if*

$$m \gtrsim \tau^4 \left(w(\mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}}) \cap \mathbb{S}^{d-1}) + \sqrt{\log(\eta^{-1})} \right)^2 .$$

Surprisingly, this bound suggests almost the same scaling behavior as in the Gaussian case (cf. Corollary 1), barring the dependence on the subgaussian parameter τ, as well as an absolute constant hidden in the notation.

The results mentioned so far are not without their own set of drawbacks. While robustness against noise was established in Proposition 2, the tangent cone characterization is inherently susceptible to model deficiencies. For instance, consider again the example $\mathcal{A} = \{\pm \mathbf{e}_i\}$ giving rise to the set of $\Sigma_k(\mathbb{R}^d)$. If $\mathring{\mathbf{x}}$ is not a sparse linear combination of elements in $\mathcal{A}$ (e.g., $\mathring{\mathbf{x}}$ may only be compressible rather than exactly sparse), then the tangent cone of $\|\cdot\|_{\mathcal{A}}$ at $\mathring{\mathbf{x}}$ may not have a small mean width at all as we saw in Fig. 2. In fact, in this case, $w(\mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}}) \cap \mathbb{S}^{d-1})^2$ is usually on the order of the ambient dimension d [80]. Moreover, as we also demonstrated graphically in Fig. 2, the recovery guarantees presented in this section only apply to individual vectors. Such results are customarily referred to as nonuniform guarantees in the compressed sensing literature. Before moving on to the uniform recovery case which provides recovery conditions for *all* vectors in a signal class simultaneously, we want to briefly comment on an important line of work connecting sparse recovery with the field of conic integral geometry. This is the subject of the next section.

4.2 Connections to Conic Integral Geometry

In an independent line of research [4], the sparse recovery problem was recently approached from the perspective of conic integral geometry. At the heart of this field lies the study of the so-called *intrinsic volumes of cones*. We limit our discussion to the important class of polyhedral cones[10] here, and refer interested readers to [4] for a treatment of general convex cones.

Definition 7 (*Intrinsic volumes*) Let $\mathcal{C}$ be a polyhedral cone in $\mathbb{R}^d$, and denote by $\mathbf{g}$ a standard Gaussian random vector. Then for $i = 0, \dots, d$, the ith intrinsic volume of $\mathcal{C}$ is defined as

$$v_i(\mathcal{C}) := \mathbb{P}(\Pi_{\mathcal{C}}(\mathbf{g}) \in \mathcal{F}_i(\mathcal{C})),$$

where $\Pi_{\mathcal{C}}$ denotes the orthogonal projector on $\mathcal{C}$, and $\mathcal{F}_i(\mathcal{C})$ denotes the union of relative interiors of all i-dimensional faces of $\mathcal{C}$.

If we are given two non-empty convex cones $\mathcal{C}, \mathcal{D} \subset \mathbb{R}^d$, one of which is not a subspace, and we draw an orthogonal matrix $\mathbf{Q} \in \mathbb{R}^{d \times d}$ from the uniform Haar measure, then the probability that $\mathcal{C}$ and the randomly rotated cone $\mathbf{Q}\mathcal{D}$ intersect nontrivially is fully determined by the intrinsic volumes of $\mathcal{C}$ and $\mathcal{D}$. The precise statement of this result is known as the *conic kinematic formula*.

Theorem 4 (Conic kinematic formula, [4, Fact 2.1]) *Let $\mathcal{C}$ and $\mathcal{D}$ be two non-empty closed convex cones in $\mathbb{R}^d$ of which at most one is a subspace. Denote by $\mathbf{Q} \in \mathrm{O}(d)$ a matrix drawn uniformly from the Haar measure on the orthogonal group. Then*

$$\mathbb{P}(\mathcal{C} \cap \mathbf{Q}\mathcal{D} \neq \{\mathbf{0}\}) = \sum_{i=0}^{d} (1 + (-1)^{i+1}) \sum_{j=1}^{d} v_i(\mathcal{C}) v_{d+i-j}(\mathcal{D}).$$

To apply this result to the context of sparse recovery as discussed in the previous section, one simply chooses $\mathcal{C} = \mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}})$, and $\mathcal{D} = \ker(\mathbf{A})$, similar to the situation of Gordon's escape theorem. While the intrinsic volumes of $\ker(\mathbf{A})$, a $(d - m)$-dimensional linear subspace, are easily determined by[11]

$$v_i(\ker(\mathbf{A})) = \begin{cases} 1, & i = d - m, \\ 0, & \text{otherwise}, \end{cases} \tag{16}$$

the calculation of the intrinsic volumes of tangent cones is much less straightforward. Fortunately, there is an elegant way out of this situation which was first demonstrated

[10] A cone $\mathcal{C} \subset \mathbb{R}^d$ is called *polyhedral* if it can be expressed as the intersection of finitely many half-spaces.

[11] This follows from the fact that $\ker(\mathbf{A})$ only has a single face on which $\Pi_{\ker(\mathbf{A})}$ projects every point $\mathbf{x} \in \mathbb{R}^d$, namely, $\ker(\mathbf{A})$ itself.

in [4]. Since any vector $\mathbf{x} \in \mathbb{R}^d$ projected on a closed convex cone $\mathcal{C}$ must belong to exactly one of the $d+1$ sets $\mathcal{F}_i(\mathcal{C})$ defined in Definition 7, the collection $\{v_i(\mathcal{C})\}_{i=0}^d$ of intrinsic volumes defines a discrete probability distribution on $\{0, 1, \ldots, d\}$. Moreover, the distribution can be shown to concentrate sharply around its expectation

$$\delta(\mathcal{C}) := \sum_{i=0}^{d} i v_i(\mathcal{C}),$$

known as the *statistical dimension* of $\mathcal{C}$, which in turn can be tightly estimated in many cases of interest by appealing to techniques from convex analysis. In fact, the same technique was previously used in [31] to derive tight estimates of the mean width of various tangent cones. Note, however, that this work merely exploited a numerical relation between the Gaussian mean width and the statistical dimension which we will comment on below but was not generally motivated by conic integral geometry. The concentration behavior of intrinsic volumes ultimately allowed Amelunxen et al. to derive the following remarkable pair of bounds which constitute a breakthrough result in the theory of sparse recovery.

Theorem 5 (Approximate conic kinematic formula, [4, Theorem II]) *Let* $\mathring{\mathbf{x}} \in \operatorname{cone}_k(\mathcal{A})$*, and denote by* $\mathbf{A} \in \mathbb{R}^{m \times d}$ *a standard Gaussian matrix with independent entries as usual. Given the linear observations* $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}}$*, and denoting by* $\mathbf{x}^\star$ *the optimal solution of Problem* (8)*, the following two statements hold for* $\eta \in (0, 1]$:

$$\begin{aligned} \mathbb{P}(\mathbf{x}^\star = \mathring{\mathbf{x}}) \geq 1 - \eta \quad &\textit{if} \quad m \geq \delta(\mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}})) + c_\eta\sqrt{d}, \\ \mathbb{P}(\mathbf{x}^\star \neq \mathring{\mathbf{x}}) \leq \eta \quad &\textit{if} \quad m \leq \delta(\mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}})) - c_\eta\sqrt{d} \end{aligned}$$

with $c_\eta = \sqrt{8\log(4/\eta)}$.

Before addressing the problem of estimating the statistical dimension δ of the tangent cone $\mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}})$, let us briefly comment on the above result first. Theorem 5 is remarkable for a variety of reasons. First, as was demonstrated numerically in [4], the two bounds correctly predict the position of the so-called phase transition. Such results were previously only known in the asymptotic large-system limit (cf. [43, 45]) where one considers for $d, m, k \to \infty$ the fixed ratios $\delta := m/d$, and $\rho := k/m$ over the open unit square $(0, 1)^2$. The phase-transition phenomenon describes a particular behavior of the system which exhibits a certain critical line $\rho^\star = \rho^\star(\delta)$ that partitions $(0, 1)^2$ into two distinct regions: one where recovery almost certainly succeeds, and one where it almost certainly fails. The transition line then corresponds to the 50th percentile. Second, it represents the first non-asymptotic result which correctly predicts a fundamental limit below which sparse recovery will fail with high probability. This is in stark contrast to previous results based on Gordon's escape theorem which were only able to predict that recovery would succeed above a certain threshold but could not make any assessment of the behavior below it. Finally, as a result of the second point, Theorem 5 represents the first result which quantifies the width of the transition region where the probability of exact recovery will change from almost certain failure to almost certain success. Once again we refer interested readers to the excellent exposition [4], particularly Sect. 10, for a thorough comparison of their results to the pertinent literature on the existence of phase transitions in compressed sensing.

The key ingredient in the application of Theorem 5 is the statistical dimension δ of the tangent cone $\mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}})$. As mentioned above, the statistical dimension is defined as the expected value of the distribution defined by the intrinsic volumes of $\mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}})$. However, it admits two alternative representations which can be leveraged to estimate $\delta(\mathcal{C})$, especially when $\mathcal{C}$ corresponds to a tangent cone. This is the content of the following result.

Proposition 3 (Statistical dimension, [4, Proposition 3.1]) *Let $\mathcal{C}$ be a closed convex cone in $\mathbb{R}^d$, and let $\mathbf{g}$ be a standard Gaussian d-vector. Then*

$$\delta(\mathcal{C}) = \sum_{i=0}^{d} i v_i(\mathcal{C}) = \mathbb{E}\big[\|\Pi_{\mathcal{C}}(\mathbf{g})\|_2^2\big] = \mathbb{E}\big[\operatorname{dist}(\mathbf{g}, \mathcal{C}^\circ)^2\big],$$

where $\mathcal{C}^\circ := \big\{\mathbf{z} \in \mathbb{R}^d : \langle \mathbf{x}, \mathbf{z}\rangle \le 0 \ \forall \mathbf{x} \in \mathcal{C}\big\}$ denotes the polar cone of $\mathcal{C}$.

In particular, we want to focus on the last identity when $\mathcal{C} = \mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}})$. In fact, in this situation one may exploit a well-known fact from convex geometry that states that the polar cone of the tangent cone corresponds to the normal cone [88]

$$\begin{aligned}\mathcal{N}_{\mathcal{A}}(\mathring{\mathbf{x}}) &:= \big\{\mathbf{v} \in \mathbb{R}^d : \big\langle \mathbf{v}, \mathbf{x} - \mathring{\mathbf{x}}\big\rangle \le 0 \ \forall \mathbf{x}\colon\ \|\mathbf{x}\|_{\mathcal{A}} \le \big\|\mathring{\mathbf{x}}\big\|_{\mathcal{A}}\big\} \\ &= \big\{\mathbf{v} \in \mathbb{R}^d : \langle \mathbf{v}, \mathbf{d}\rangle \le 0 \ \forall \mathbf{d} \in \mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}})\big\},\end{aligned}$$

which in turn can be expressed as the conic hull of the subdifferential of the atomic norm at $\mathring{\mathbf{x}}$,

$$\mathcal{T}_{\mathcal{A}}(\mathring{\mathbf{x}})^\circ = \mathcal{N}_{\mathcal{A}}(\mathring{\mathbf{x}}) = \operatorname{cone}(\partial \big\|\mathring{\mathbf{x}}\big\|_{\mathcal{A}}) = \bigcup_{t \ge 0} t\partial \big\|\mathring{\mathbf{x}}\big\|_{\mathcal{A}}.$$

The last identity follows from the fact that the subdifferential of a convex function is always a convex set. In other words, given a recipe for the subdifferential of the atomic norm, the statistical dimension of its associated tangent cone can be estimated by bounding the expected distance of a Gaussian vector to its convex hull. In many cases of interest, this turns out to be a comparatively easy task (see, e.g., [31, Appendix C], [55, Appendix A] and [4, Sect. 4]).

As alluded to before, the statistical dimension also shares a close connection to the Gaussian mean width. In particular, we have the following two inequalities (cf. [4, Proposition 10.2]):

$$w(\mathcal{C} \cap \mathbb{S}^{d-1})^2 \le \mathbb{E}\big[\operatorname{dist}(\mathbf{g}, \mathcal{C}^\circ)^2\big] = \delta(\mathcal{C}) \le w(\mathcal{C} \cap \mathbb{S}^{d-1})^2 + 1.$$

This shows that estimating the mean width is qualitatively equivalent to estimating δ. As previously mentioned, this connection was used in [31] to derive precise bounds for the mean widths of the tangent cones for sparse vectors, and low-rank matrices, as well as for block- and group-sparse signals in [55] and [84], respectively. Note that the connection between mean width and statistical dimension was already used in the pioneering works of Stojnic [91], as well as Oymak and Hassibi [76], even if the term *statistical dimension* was originally coined in [4] where the connection between the probability distribution induced by the intrinsic volumes and its projective characterization in Proposition 3 was first established. We want to emphasize again that the fundamental significance of the statistical dimension in the context of sparse recovery did not become clear until the seminal work of Amelunxen, Lotz, McCoy, and Tropp who rigorously demonstrated the concentration behavior of intrinsic volumes, culminating in the breakthrough result stated in Theorem 5. In the same context, the authors argued that the statistical dimension generally represents a more appropriate measure of "dimension" of cones than the mean width. For instance, if $\mathcal{C}$ is an n-dimensional linear subspace L_n of $\mathbb{R}^d$, then it immediately follows from Eq. (16) that $\delta(L_n) = \dim(L_n) = n$. Moreover, given a closed convex cone $\mathcal{C} \subset \mathbb{R}^d$, we have $\delta(\mathcal{C}) + \delta(\mathcal{C}^\circ) = d$ (cf. [4, Proposition 3.1]) which generalizes the property $\dim(L_n) + \dim(L_n^\perp) = d$ from linear subspaces to convex cones since $L_n^\circ = L_n^\perp$, i.e., the polar cone of a subspace is its orthogonal complement.

The concepts discussed in this section all addressed the problem of recovering or estimating individual vectors with a low-complexity structure from low-dimensional linear measurements. In other words, given two vectors $\mathring{\mathbf{x}}$ and $\mathring{\mathbf{x}}'$ with the same low-complexity structure, and the knowledge that $\mathring{\mathbf{x}}$ can be estimated with a particular accuracy, we are not able to infer that the same accuracy also holds when we try to recover $\mathring{\mathbf{x}}'$ given a fixed choice of $\mathbf{A}$. Recall, for example, the situation illustrated in Fig. 2a. If instead of $\mathring{\mathbf{x}}$ we observe a vector $\mathring{\mathbf{x}}'$ positioned on the rightmost vertex of the scaled ℓ_1-ball, the tangent cone at $\mathring{\mathbf{x}}'$ now corresponds to the tangent cone at $\mathring{\mathbf{x}}$ rotated 90° clockwise around the origin. However, since this cone intersects the null space of $\mathbf{A}$ at arbitrarily many points, we are not able to recover $\mathring{\mathbf{x}}$ and $\mathring{\mathbf{x}}'$ simultaneously. In the parlance of probability theory, we might say that the results presented in this section are conditioned on a particular choice of $\mathring{\mathbf{x}}$. Such results are therefore known as *nonuniform* guarantees as they do not hold uniformly for all signals in a particular class at once.

In contrast, in the next section, we will introduce a variety of properties of measurement matrices which will allow us to characterize the recovery behavior uniformly over all elements in a signal class given the same choice of measurement matrix. Most importantly, we will focus on a particularly important property which not only yields a sufficient condition for perfect recovery of sparse vectors but one which has also proven an indispensable tool in providing stability and robustness conditions in situations where we are tasked with the recovery of signals from corrupted measurements.

5 Exact Recovery of Sparse Vectors

In this section, we consider conditions under which the sparse linear inverse problem, in which we are to infer a d-dimensional vector $\mathring{\mathbf{x}} \in \Sigma_k$ from its linear measurements $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}} \in \mathbb{R}^m$, admits a unique solution. In contrast to the situation discussed in Sect. 4, we are now specifically interested in conditions under which the entire set Σ_k can be recovered or at least well approximated by a single measurement matrix $\mathbf{A}$.

Consider two vectors $\mathbf{x}, \mathbf{z} \in \Sigma_k$, and suppose that both vectors are mapped to the same point $\mathbf{y} = \mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{z}$ such that $\mathbf{x} - \mathbf{z} \in \ker(\mathbf{A})$. Obviously, unless we specifically ask that $\mathbf{x} \neq \mathbf{z}$, there is absolutely no chance that we would ever be able to decide which element in Σ_k generated the measurements $\mathbf{y}$. In other words, if there is to be any hope to ever uniquely identify sparse vectors from their image under $\mathbf{A}$, the most fundamental condition we must impose is that no two vectors in Σ_k are mapped to the same point $\mathbf{y}$ in $\mathbb{R}^m$. However, since the difference of two k-sparse vectors is $2k$-sparse, this immediately yields the condition $\ker(\mathbf{A}) \cap \Sigma_{2k} = \{\mathbf{0}\}$. In words, the linear inverse problem for sparse vectors is well-posed if and only if the only $2k$-sparse vector contained in the null space of $\mathbf{A}$ is the zero vector.

Note that this viewpoint differs from the way we approached the recovery problem earlier in Sect. 4 where we merely asked for a particular optimization problem defined in terms of a fixed vector $\mathring{\mathbf{x}} \in \mathcal{K}$ to have a unique solution which ultimately lead us to the local tangent cone condition in Proposition 1. This also explains why, in the example depicted in Fig. 2a, we were able to recover the 1-sparse vector $\mathring{\mathbf{x}} \in \mathbb{R}^2$ but not the 1-sparse vector $\mathring{\mathbf{x}}'$. As the considerations above show, there simply is no circumstance under which we would ever be able to uniquely recover every 1-sparse vector in $\mathbb{R}^2$ from scalar measurements $y \in \mathbb{R}$. This is due to the fact that the null space of any matrix $\mathbf{A} \in \mathbb{R}^{1\times 2}$ (a row vector) either corresponds to a line through the origin or the entire plane $\mathbb{R}^2$ itself if $\mathbf{A} = \mathbf{0}$. However, since the set of 2-sparse vectors in $\mathbb{R}^2$ also corresponds to $\mathbb{R}^2$, the subspace $\ker(\mathbf{A})$ intersects Σ_2 at arbitrarily many points regardless of the choice of $\mathbf{A}$, violating the condition $\ker(\mathbf{A}) \cap \Sigma_2 = \{\mathbf{0}\}$.

The following theorem, which constitutes a key result in compressed sensing, formalizes the observations above.

Theorem 6 ([54, Theorem 2.13]) *Given a matrix* $\mathbf{A} \in \mathbb{C}^{m\times d}$, *the following statements are equivalent:*

1. *Given a vector* $\mathring{\mathbf{x}} \in \mathbb{C}^d$ *supported on a set of size at most k, the problem*

$$\begin{aligned} &\textit{minimize} \quad \|\mathbf{x}\|_0 \\ &\textit{s.t.} \qquad \mathbf{A}\mathring{\mathbf{x}} = \mathbf{A}\mathbf{x} \end{aligned} \tag{P$_0$}$$

   has a unique k-sparse minimizer, namely, $\mathbf{x}^\star = \mathring{\mathbf{x}}$.
2. *Every vector* $\mathring{\mathbf{x}}$ *is the unique k-sparse solution of the system* $\mathbf{A}\mathbf{z} = \mathbf{A}\mathring{\mathbf{x}}$.
3. *The only 2k-sparse vector contained in the null space of* $\mathbf{A}$ *is the zero vector, i.e.,* $\ker(\mathbf{A}) \cap \Sigma_{2k} = \{\mathbf{0}\}$.

The key insight of the above result is the equivalence between the condition $\ker(\mathbf{A}) \cap \Sigma_{2k} = \{\mathbf{0}\}$, and the existence of sparse minimizers of a particularly important nonconvex optimization problem. More precisely, we have by Theorem 1 that a natural strategy to recover a sparse vector $\mathring{\mathbf{x}} \in \Sigma_k$ given $\mathbf{y}$ and $\mathbf{A}$ corresponds to a search for the sparsest element in the affine space $\left\{\mathbf{x} \in \mathbb{C}^d : \mathbf{A}\mathbf{x} = \mathbf{y}\right\}$.

One immediate question arising from Theorem 6 is "how underdetermined" the system $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}}$ is allowed to become for there to still be a unique solution. Remarkably, Problem (P_0) can be shown to uniquely recover the original vector $\mathring{\mathbf{x}}$ as soon as the rank of the measurement matrix $\mathbf{A} \in \mathbb{C}^{m \times d}$ exceeds the critical threshold $\operatorname{rank} \mathbf{A} \geq 2k$ [54]. In other words, every $2k$ columns of $\mathbf{A}$ must be linearly independent. Motivated by this observation, some authors refer to the so-called spark of a matrix—a portmanteau of the words "sparse" and "rank"—as the smallest number of linearly dependent columns of $\mathbf{A}$ [41]. With this definition, the rank constraint can be equivalently stated as $\operatorname{spark}(\mathbf{A}) > 2k$. Given a measurement matrix $\mathbf{A}$ of size $m \times d$ in the regime $m < d$, perfect recovery of any k-sparse vector is therefore guaranteed as soon as $\operatorname{spark} \mathbf{A} > 2k$. Moreover, since $\operatorname{rank}(\mathbf{A}) \leq m$, the rank requirement $\operatorname{rank}(\mathbf{A}) \geq 2k$ ultimately yields the necessary condition $m \geq 2k$ for perfect recovery of all k-sparse vectors via ℓ_0-minimization.

As alluded to before, an important distinction between the rank characterization above, and the tangent cone condition from Proposition 1 is that the latter only applies to individual elements of Σ_k while the requirement $\operatorname{rank}(\mathbf{A}) \geq 2k$ implies perfect recovery of every k-sparse vector via ℓ_0-minimization. If we are only interested in a nonuniform recovery condition, it turns out that we already get by with $m \geq k + 1$ measurements [54, Sect. 2.2]. Note, however, that the condition in Proposition 1 is based on a tractable optimization problem. This stands in stark contrast to the ℓ_0-minimization problem (P_0) which is provably NP-hard as it can be reduced to the so-called *exact 3-set cover* problem which in turn is known to belong to the class of NP-complete problems [72]. As a result, solving Problem (P_0) requires a combinatorial search over all $\sum_{i=0}^{d} \binom{d}{i}$ possible subproblems if k is unknown and $\binom{d}{k}$ otherwise, both of which are intractable for even moderately sized problems. While there exist certain deterministic matrices which satisfy the rank condition such as Vandermonde matrices, as well as tractable algorithms such as *Prony's method* to solve the associated ℓ_0-minimization problem, the solution of the general problem remains out of reach unless $\mathrm{P} = \mathrm{NP}$. Moreover, another drawback of attempting to solve the ℓ_0-minimization problem directly is that it can be shown to be highly sensitive to measurement noise and sparsity defects [54, Chap. 2].

While Theorem 6 in and of itself already represents a fascinating result in the field of linear algebra, the story does not end there. Despite the seemingly dire situation we find ourselves in when attempting to find minimizers of Problem (P_0), one of the key insights in the theory of compressed sensing is that there is a convenient escape hatch in the form of convex relaxations. In fact, it turns out that under slightly more demanding conditions on the null space of $\mathbf{A}$, we are still able to faithfully recover sparse or approximately sparse vectors by turning to a particular relaxation of Problem (P_0). We are, of course, talking about the infamous ℓ_1-minimization problem which we already discussed implicitly in the context of atomic norm minimization

w. r. t. the atomic set $\mathcal{A} = \{\pm \mathbf{e}_i\}$ generating the set of sparse vectors. It is this insight which elevates the field of compressed sensing from a purely mathematical theory to a highly desirable tool with far-reaching implications in countless domains of engineering, physics, chemistry, and biology. Before discussing the particular conditions on $\mathbf{A}$ which allow for robust and most importantly efficient recovery of sparse vectors from underdetermined linear measurements, let us first state and briefly comment on what is by now probably one of the most well-known and well-studied optimization problems in mathematics to date.

In light of our discussion of compressible vectors in Sect. 3.1, the following optimization problem, famously known as the basis pursuit (BP) problem, naturally represents the closest convex relaxation of the nonconvex ℓ_0-minimization problem (P_0):

$$\begin{aligned} \underset{\mathbf{x}}{\text{minimize}} \quad & \|\mathbf{x}\|_1 \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} = \mathbf{A}\mathring{\mathbf{x}}. \end{aligned} \tag{P_1}$$

Ignoring for a moment any structural properties on the vector $\mathring{\mathbf{x}}$ we aim to recover, as well as the properties of the measurement matrix $\mathbf{A} \in \mathbb{C}^{m \times d}$, the program can be shown to yield m-sparse minimizers [54, Theorem 3.1]. This observation alone already serves as a strong indicator of the deep connection between ℓ_1-minimization and sparse recovery. Moreover, the relaxation can be solved in polynomial time by so-called interior-point methods, a class of algorithms which is by now considered a standard tool in the field of convex optimization. In particular, in the real setting, Problem (P_1) belongs to the class of linear programs (LPs), while in the complex case the problem can be transformed into a second-order cone program (SOCP) over the Cartesian product of d Lorentz cones $\mathcal{K}_\mathrm{L} := \left\{(\mathbf{z}, t) \in \mathbb{R}^2 \times \mathbb{R}_{\geq 0} : \|\mathbf{z}\|_2 \leq t\right\}$.

6 Characterization of Measurement Matrices

At the beginning of Sect. 4, we presented a necessary and sufficient condition for the exact recovery of vectors in simple sets from underdetermined linear measurements (cf. Proposition 1). This condition is very much local in nature as it depends on the particular choice of the vector one aims to recover. To circumvent this issue, we turned to random matrices which allowed us to draw on powerful probabilistic methods to bound the probability that, conditioned on the choice of a particular vector, we would be able to recover it via atomic norm minimization.

It turns out that in a sense, this strategy can be mirrored in the case of uniform recovery of sparse vectors. However, rather than directly estimating the probability that the condition in Theorem 3 as established in the previous section holds for a particular choice of random matrix, we first introduce a few common properties of general measurement matrices, some of which will enable us to state powerful recovery guarantees which hold over entire signal classes rather than individual vectors. In Sect. 7, we will then present a series of results which assert that for many different choices of random measurement ensembles, such properties can be shown to be satisfied with overwhelmingly high probability, provided the number of measurements is chosen appropriately.

6.1 Null Space Property

As alluded to before, the relaxation of the original ℓ_0-minimization problem to a tractable convex program comes at the price of a critical difference to Problem (P$_0$). While the only requirement for Problem (P$_0$) to recover the original vector $\mathring{\mathbf{x}} \in \Sigma_k$ was for the number of measurements to exceed $2k$, perfect recovery will now be dependent on a certain structural property of the null space of $\mathbf{A}$, aptly referred to as the null space property (NSP), which was first introduced in [33].

Definition 8 (*Null space property*) A matrix $\mathbf{A} \in \mathbb{C}^{m\times d}$ is said to satisfy the NSP of order k if, for any set $S \subset [d]$ with $|S| \leq k$, we have

$$\|\mathbf{v}_S\|_1 < \left\|\mathbf{v}_{\overline{S}}\right\|_1 \quad \forall \mathbf{v} \in \ker \mathbf{A}\backslash\{\mathbf{0}\}.$$

The definition of the null space property admits a few additional observations for vectors in the null space of $\mathbf{A}$. Consider again an index set $S \subset [d]$ of size at most k. Then for $\mathbf{v} \in \ker \mathbf{A} \setminus \{\mathbf{0}\}$ we have

$$\begin{aligned}\|\mathbf{v}\|_1 = \left\|\mathbf{v}_S + \mathbf{v}_{\overline{S}}\right\|_1 &= \|\mathbf{v}_S\|_1 + \left\|\mathbf{v}_{\overline{S}}\right\|_1 \\ &< \left\|\mathbf{v}_{\overline{S}}\right\|_1 + \left\|\mathbf{v}_{\overline{S}}\right\|_1 \\ &= 2\left\|\mathbf{v}_{\overline{S}}\right\|_1.\end{aligned}$$

Moreover, if S is the set supporting the largest components of $\mathbf{v}$ in absolute value, one has with the definition of the best k-term approximation error in Eq. (5),

$$\|\mathbf{v}\|_1 < 2\sigma_k(\mathbf{v})_1.$$

Finally, by the Cauchy–Schwarz inequality, we have that for any $\mathbf{v} \in \mathbb{C}^d$, it holds that $\|\mathbf{v}\|_1^2 \leq \|\mathbf{v}\|_0 \cdot \|\mathbf{v}\|_2^2$. Therefore, one often alternatively finds the condition

$$\|\mathbf{v}_S\|_2 < \frac{1}{\sqrt{k}}\left\|\mathbf{v}_{\overline{S}}\right\|_1$$

in the definition of the null space property.

Given a matrix that satisfies the null space property, we can now state the general result for the recovery of any k-sparse vector $\mathring{\mathbf{x}} \in \mathbb{C}^d$ from its linear measurements by solving the basis pursuit (BP) problem (BP) below. In particular, consider a vector $\mathbf{v} \in \ker \mathbf{A} \cap \Sigma_{2k}$ supported on an index set $S \subset [d]$ of size $2k$, and assume further that $\mathbf{v} \neq \mathbf{0}$. Then for two disjoint sets $S_1, S_2 \subset S$ with $S = S_1 \cup S_2$ and $|S_1| = |S_2| = k$, by the null space property we have $\|\mathbf{v}_{S_1}\|_1 < \|\mathbf{v}_{\overline{S_1}}\|_1 = \|\mathbf{v}_{S \setminus S_1}\|_1 = \|\mathbf{v}_{S_2}\|_1$ and $\|\mathbf{v}_{S_2}\|_1 < \|\mathbf{v}_{S_1}\|_1$ which is a contradiction, and hence $\mathbf{v} = \mathbf{0}$. In other words, the null space property implies that the null space of $\mathbf{A}$ only contains a single $2k$-sparse vector: the zero vector. This implies the condition we previously stated in Theorem 3 which said that ℓ_0-minimization can recover any k-sparse vector as long as the null space of the measurement matrix contains no $2k$-sparse vectors save for the zero vector. Amazingly, the null space property provides a necessary and sufficient condition for the following recovery guarantee for sparse vectors.

Theorem 7 *Let $\mathbf{A} \in \mathbb{C}^{m \times d}$ and $k \in [d]$. Then every k-sparse vector $\mathring{\mathbf{x}}$ is the unique minimizer of the basis pursuit problem*

$$\begin{aligned} &\textit{minimize} \ \|\mathbf{x}\|_1 \\ &\quad \textit{s.t.} \qquad \mathbf{y} = \mathbf{A}\mathbf{x} \end{aligned} \tag{BP}$$

with $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}}$ iff $\mathbf{A}$ satisfies the null space property of order k.

Proof If $\mathbf{A}\mathring{\mathbf{x}} = \mathbf{A}\mathbf{z}$, then $\mathbf{d} := \mathring{\mathbf{x}} - \mathbf{z} \in \ker(\mathbf{A})$ with $\mathbf{d}_S = \mathring{\mathbf{x}} - \mathbf{z}_S$ and $\mathbf{d}_{\overline{S}} = \mathbf{z}_{\overline{S}}$. Invoking the null space property we have

$$\begin{aligned} \left\|\mathring{\mathbf{x}}\right\|_1 &= \left\|\mathring{\mathbf{x}} - \mathbf{z}_S + \mathbf{z}_S\right\|_1 \\ &\leq \|\mathbf{d}_S\|_1 + \|\mathbf{z}_S\|_1 \\ &< \left\|\mathbf{d}_{\overline{S}}\right\|_1 + \|\mathbf{z}_S\|_1 \\ &= \left\|\mathbf{z}_{\overline{S}}\right\|_1 + \|\mathbf{z}_S\|_1 = \|\mathbf{z}\|_1 . \end{aligned}$$

This means that $\mathring{\mathbf{x}}$ is the unique minimizer of (BP). For the other direction, every $\mathbf{v} \in \ker(\mathbf{A})$ satisfies $\mathbf{A}\mathbf{v}_S = \mathbf{A}(-\mathbf{v}_{\overline{S}})$. Since $\mathbf{v}_S$ is the unique minimizer of (BP), we have $\|\mathbf{v}_S\|_1 < \| - \mathbf{v}_{\overline{S}}\|_1$ which is the null space property. □

Two situations are of particular importance in linear inverse problems, namely, situations in which $\mathring{\mathbf{x}}$ is only approximately sparse, and when the measurements are corrupted by additive noise. It is therefore generally desirable for a recovery algorithm to be both robust to noise and stable w.r.t. to so-called sparsity defect. To that end, one can extend the definition of the null space property to provide similar guarantees to the one stated in Theorem 7. We first consider the so-called stable null space property which can be used to account for sparsity defects of vectors.

Definition 9 (*Stable null space property*) A matrix $\mathbf{A} \in \mathbb{C}^{m \times d}$ is said to satisfy the stable null space property of order k with constant $0 < \rho < 1$ w.r.t. any set $S \subset [d]$ if

$$\|\mathbf{v}_S\|_1 \leq \rho \left\|\mathbf{v}_{\overline{S}}\right\|_1 \quad \forall \mathbf{v} \in \ker \mathbf{A}$$

with $|S| \leq k$.

With this definition in place, the following result characterizes the impact of sparsity defects on the recovery error of the basis pursuit problem.

Theorem 8 ([54, Theorem 4.12]) *Let* $\mathbf{A} \in \mathbb{C}^{m \times d}$ *and* $k \in [d]$. *Then with* $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}}$, *the solution* $\mathbf{x}^\star$ *of Problem* (BP) *satisfies*

$$\left\|\mathbf{x}^\star - \mathring{\mathbf{x}}\right\|_2 \leq \frac{2(1+\rho)}{(1-\rho)} \sigma_k(\mathring{\mathbf{x}})_1$$

if $\mathbf{A}$ *satisfies the stable null space property of order* k. *In particular, if* $\mathring{\mathbf{x}} \in \Sigma_k$ *then* $\mathbf{x}^\star = \mathring{\mathbf{x}}$.

We can extend the definition of the stable null space property once more to also account for additive noise in the measurements. For reference, we state here the most general form of the so-called ℓ_q-robust null space property. However, instead of using this definition to state a stable, noise-robust counterpart to Theorem 8, we will instead turn to a more commonly used property of measurement matrices in the next section to state a guarantee of this type.

Definition 10 (*ℓ_q-robust null space property*) Let $q \geq 1$, and denote by $\|\cdot\|$ an arbitrary norm on $\mathbb{C}^m$. Then the matrix $\mathbf{A} \in \mathbb{C}^{m \times d}$ satisfies the ℓ_q-robust null space property of order k with constants $0 < \rho < 1$ and $\tau > 0$ if for all $\mathbf{v} \in \mathbb{C}^d$,

$$\|\mathbf{v}_S\|_q \leq \frac{\rho}{k^{1-1/q}} \left\|\mathbf{v}_{\overline{S}}\right\|_1 + \tau \|\mathbf{A}\mathbf{v}\|$$

for all $S \subset [d]$, $|S| \leq k$.

Theorem 7 yields a necessary and sufficient condition for the matrix $\mathbf{A}$ that answers the central question when minimizers of (P_0) and (P_1) coincide. While this represents an invaluable result, Theorem 7 makes no statement regarding the actual existence of such matrices. As it turns out, constructing deterministic matrices which directly satisfy the null space property (or its stable or noise-robust variants) constitutes a highly nontrivial problem. In fact, even verifying whether a given matrix satisfies the null space property was eventually shown to be an NP-hard decision problem [95]. Fortunately, it can be shown that matrices satisfying the null space property still exist in abundance if one turns to random measurement ensembles. While it is possible to directly establish the existence of such matrices probabilistically,[12] it has become common practice in the compressed sensing literature to mainly consider an alternative property of measurement matrices to establish recovery guarantees. The property in question is of course the infamous restricted isometry property (RIP) which was introduced in one of the very first papers on compressed sensing [27], and by now constitutes one of the most well-studied objects in the theory.

[12] In fact, as we will briefly discuss in Sect. 7, such random constructions are often characterized by more well-behaved scaling constants.

6.2 Restricted Isometry Property

The restricted isometry property (RIP) was first introduced in the seminal work by Candes and Tao [27], and shown in [21] to allow for robust recovery of approximately sparse vectors in the presence of measurement noise. While this property only yields a sufficient condition implying the null space property, matrices of this type can be found—at least in a probabilistic sense—in abundance as various random measurement ensembles can be shown to satisfy the RIP with high probability (cf. Sect. 7). The property is defined as follows.

Definition 11 A matrix $\mathbf{A} \in \mathbb{C}^{m \times d}$ is said to satisfy the RIP of order k if

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

for all $\mathbf{x} \in \Sigma_k$ with $\delta \geq 0$. The smallest $\delta_k \leq \delta$ satisfying this condition is called the restricted isometry constant (RIC) of $\mathbf{A}$.

Intuitively, this definition states that for any $S \subset [d]$ with $|S| \leq k$ the submatrix $\mathbf{A}_S$ obtained by retaining only the columns indexed by S approximately acts like an isometry on the set of k-sparse vectors which admits an alternative characterization of the restricted isometry constant δ_k as

$$\delta_k = \max_{\substack{S \subset [d], \\ |S| = k}} \left\| \mathbf{A}_S^* \mathbf{A}_S - \mathrm{Id} \right\|_{2 \to 2}.$$

This definition of the restricted isometry constant is commonly used in proofs establishing the restricted isometry property in a probabilistic setting by showing that δ_k concentrates sharply around its expectation.

In light of the importance and popularity of the restricted isometry property in compressed sensing, we will state most recovery conditions of the various algorithms introduced in Sect. 8 exclusively in terms of the restricted isometry constants associated with the RIP matrices in question.

The restricted isometry property admits a particularly short and concise proof of why k-sparse vectors have unique measurement vectors $\mathbf{y}$ under projections through $\mathbf{A}$. Assume the matrix $\mathbf{A} \in \mathbb{C}^{m \times d}$ satisfies the RIP condition of order $2k$ with constant $\delta_{2k} < 1$, and consider two distinct k-sparse vectors $\mathbf{x}, \mathbf{z} \in \mathbb{C}^d$ with $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{z}$. Define now $\mathbf{v} := \mathbf{x} - \mathbf{z} \in \Sigma_{2k}$, i.e., $\mathbf{A}\mathbf{v} = \mathbf{0}$. Then we have by the restricted isometry property,

$$0 < (1 - \delta_{2k}) \|\mathbf{v}\|_2^2 \leq \|\mathbf{A}\mathbf{v}\|_2^2 = 0.$$

Since this only holds for $\mathbf{v} = \mathbf{0}$, we must have $\mathbf{x} = \mathbf{z}$. In other words, if $\mathbf{A}$ is an RIP matrix of order $2k$, no two k-sparse vectors are mapped to the same measurement vector $\mathbf{y}$ through $\mathbf{A}$.

In the following, we consider noisy measurements of the form $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}} + \mathbf{e}$ where the additive noise term $\mathbf{e} \in \mathbb{C}^m$ is assumed to be bounded according to $\|\mathbf{e}\|_2 \leq \eta$.

Under assumption of the restricted isometry property, one may then establish the following stable and robust recovery result.

Theorem 9 ([54, Theorem 6.12]) *Let* $\mathbf{A} \in \mathbb{C}^{m \times d}$ *be a matrix satisfying the RIP of order* $2k$ *with restricted isometry constant* $\delta_{2k} < 4/\sqrt{41}$. *For* $\mathring{\mathbf{x}} \in \mathbb{C}^d$, *and* $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}} + \mathbf{e}$ *with* $\|\mathbf{e}\|_2 \leq \eta$, *denote by* $\mathbf{x}^\star$ *the solution of the quadratically constrained basis pursuit problem*

$$\begin{aligned} &\textit{minimize} \ \ \|\mathbf{x}\|_1 \\ &\quad \textit{s.t.} \qquad \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2 \leq \eta. \end{aligned} \tag{QCBP}$$

Then

$$\begin{aligned} \left\|\mathring{\mathbf{x}} - \mathbf{x}^\star\right\|_1 &\leq C\sigma_k(\mathring{\mathbf{x}})_1 + D\sqrt{k}\eta, \\ \left\|\mathring{\mathbf{x}} - \mathbf{x}^\star\right\|_2 &\leq \frac{C}{\sqrt{k}}\sigma_k(\mathring{\mathbf{x}})_1 + D\eta, \end{aligned}$$

where $C, D > 0$ *depend only on* δ_{2k}.

This result is both stable w. r. t. sparsity defect and robust against additive noise as the error bounds only depend on the model mismatch quantified by the best k-term approximation error of $\mathring{\mathbf{x}}$, as well as on the extrinsic noise level η. In case of exact k-sparsity of $\mathring{\mathbf{x}}$, and in the absence of measurement noise, Theorem 9 immediately implies perfect recovery.

6.3 Mutual Coherence

Despite the fact that both NSP and RIP allow for the derivation of very strong results in terms of stability and robustness of general recovery algorithms, checking either of them in practice remains an NP-hard decision problem [95]. One alternative property of a measurement matrix $\mathbf{A}$ that can easily be checked in practice is the so-called *mutual coherence*.

Definition 12 Let $\mathbf{A} \in \mathbb{C}^{m \times d}$. Then the mutual coherence $\mu = \mu(\mathbf{A})$ is defined as

$$\mu(\mathbf{A}) := \max_{1 \leq i \neq j \leq d} \frac{|\langle \mathbf{a}_i, \mathbf{a}_j \rangle|}{\|\mathbf{a}_i\|_2 \left\|\mathbf{a}_j\right\|_2},$$

where $\mathbf{a}_i$ denotes the ith column of $\mathbf{A}$. Assuming ℓ_2-normalized columns of $\mathbf{A}$, this corresponds to the largest off-diagonal element in absolute value of the Gramian $\mathbf{A}^*\mathbf{A}$ of $\mathbf{A}$.

The following proposition presents a fundamental limit on the mutual coherence of a matrix known as the *Welch bound*.

Proposition 4 ([101]) *The coherence of a matrix* $\mathbf{A} \in \mathbb{C}^{m \times d}$ *with* ℓ_2*-normalized columns satisfies*

$$\mu(\mathbf{A}) \geq \sqrt{\frac{d-m}{m(d-1)}}.$$

The equality is attained for every matrix whose columns form an equiangular tight frame.

Unfortunately, coherence-based analyses are rather pessimistic in terms of the number of measurements required to establish robust and stable recovery guarantees. In fact, it can be shown that conditions for perfect recovery in terms of the mutual coherence dictate a quadratic scaling $m = \boldsymbol{\Omega}(k^2)$ of the number of measurements [96], which is only of interest in practice at low sparsity levels.

6.4 Quotient Property

One drawback of the quadratically constrained basis pursuit (QCBP) problem (QCBP) is the fact that one has to have access to an estimate of the noise parameter $\eta \geq \|\mathbf{e}\|_2$, which is often not available in practice. Surprisingly, it can be shown, however, that under an additional condition on the measurement matrix stable and robust recovery of compressible vectors is still possible without any prior knowledge of $\|\mathbf{e}\|_2 \in \mathbb{C}^m$ by means of solving the equality-constrained basis pursuit problem. This condition is given in the form of the so-called *quotient property* of $\mathbf{A}$.

Definition 13 A matrix $\mathbf{A} \in \mathbb{C}^{m \times d}$ is said to satisfy the ℓ_1-quotient property with constant ν if for any $\mathbf{e} \in \mathbb{C}^m$ there exists a vector $\mathbf{u} \in \mathbb{C}^d$ such that

$$\mathbf{e} = \mathbf{A}\mathbf{u} \quad \text{with} \quad \|\mathbf{u}\|_1 \leq \nu\sqrt{k_*}\,\|\mathbf{e}\|_2\,,$$

where $k_* := m/\log(ed/m)$.

If a matrix satisfies both the robust null space property and the quotient property, this allows one to establish the following remarkable result.

Theorem 10 ([54, Theorem 11.12]) *Let* $\mathbf{A} \in \mathbb{C}^{m \times d}$ *be a matrix satisfying the* ℓ_2*-robust null space property as in Definition 10, as well as the* ℓ_1*-quotient property as in Definition 13. Let further* $\mathring{\mathbf{x}} \in \mathbb{C}^d$, $\mathbf{e} \in \mathbb{C}^m$, *and denote by* $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}} + \mathbf{e}$ *the noisy linear measurements of* $\mathring{\mathbf{x}}$. *Then the solution* $\mathbf{x}^\star$ *of the basis pursuit problem (*BP*) satisfies for* $k \leq ck_*$,

$$\left\|\mathring{\mathbf{x}} - \mathbf{x}^\star\right\|_2 \leq \frac{C_1}{\sqrt{k}}\sigma_k(\mathring{\mathbf{x}})_1 + C_2\,\|\mathbf{e}\|\,,$$

where $\|\cdot\|$ *denotes the norm assumed in the* ℓ_2*-robust null space property. The constants* C_1 *and* C_2 *only depend on* ρ, τ, c, *and* ν, *i.e., the parameters of the null space and quotient property, respectively.*

In the next section, we will address the construction of random measurement matrices which, with high probability, satisfy either the restricted isometry property and/or null space property, respectively. Note that similar probabilistic results can also be shown to hold for the quotient property as introduced above. However, we skip the discussion of this topic for brevity and refer interested readers to [54, Sect. 11.3] instead.

7 Probabilistic Constructions of Measurement Matrices

In this section, we present a series of results which establish the existence of suitable measurement matrices for compressed sensing in the sense that they satisfy the restricted isometry property and consequently the null space property with high probability.

7.1 Restricted Isometries

The first remarkable result we look at in this section concerns the class of subgaussian ensembles which encompasses many important instances of random measurement matrices such as Gaussian and Bernoulli matrices, as well as any matrix populated with independent copies of bounded random variables.

Theorem 11 (Subgaussian restricted isometries, [64, Theorem C.1]) *Let the rows of the* $m \times d$ *matrix* $\mathbf{A}$ *be distributed according to an independent isotropic subgaussian distribution. Then the matrix* $\frac{1}{\sqrt{m}}\mathbf{A}$ *satisfies the restricted isometry property of order* k *with constant* $\delta_k \leq \delta$ *if*

$$m \geq C\delta^{-2}k \log\left(\frac{ed}{k}\right)$$

with probability at least $1 - 2\exp(-\delta^2 m/C)$ *where the constant* C *only depends on the subgaussian norm of the rows of* $\mathbf{A}$.

A similar theorem can be stated for the case where the columns instead of rows of $\mathbf{A}$ follow a subgaussian distribution. Due to the isotropy assumption of the distribution, the random matrix $m^{-1/2}\mathbf{A}$ acts as an isometry in expectation as we would expect from an RIP matrix, i.e., $\mathbb{E}\|m^{-1/2}\mathbf{A}\mathbf{x}\|_2^2 = \|\mathbf{x}\|_2^2$. The exponential decay of the failure probability in the above theorem therefore indicates that $\|m^{-1/2}\mathbf{A}\mathbf{x}\|_2^2$ concentrates sharply around its mean $\|\mathbf{x}\|_2^2$ as intended for $\mathbf{A}$ to behave like an isometry.

The original proof of the restricted isometry property for Gaussian random matrices goes back to the work of Candès and Tao [27, 28]. As hinted at above, the restricted isometry property is usually established by means of concentration inequalities that control the deviation of $m^{-1/2}\mathbf{A}$ from its mean. In particular, such concentration results are usually based on Bernstein's inequality for subexponential random variables. In the case of Gaussian random matrices, one can appeal to slightly simpler methods that characterize the smallest and largest singular values of the Gaussian random matrices to establish the RIP in that way.

Another possible proof strategy is based on a result due to Gordon which bounds the expected minimum and maximum gain of a Gaussian random matrix acting on subsets of the sphere ([57, Corollary 1.2]). This result also lies at the heart of the proof of Gordon's escape theorem. Combined with Gaussian concentration of measure, and a simple bound on the mean width of the set of sparse vectors restricted to the unit sphere (see, for instance, [79, Lemma 2.3]), these arguments admit a simple concentration bound which implies the restricted isometry property.

Yet another proof of the restricted isometry property for Gaussian matrices is based on the famous Johnson–Lindenstrauss (JL) lemma [62] (see also [36]). Given a finite collection of points $P := \{\mathbf{x}_1, \dots, \mathbf{x}_N\} \subset \mathbb{R}^d$, and a random matrix $\mathbf{A} \in \mathbb{R}^{m \times d}$ populated with independent zero-mean Gaussian random variables with standard deviation $1/\sqrt{m}$, the JL lemma establishes a bound on the probability that the pairwise distances between the projected points $\mathbf{A}P$ and P deviate at most by a factor of $\pm\epsilon$. A matrix $\mathbf{A}$ that satisfies the property

$$(1 - \epsilon) \left\|\mathbf{x} - \mathbf{y}\right\|_2 \leq \left\|\mathbf{A}\mathbf{x} - \mathbf{A}\mathbf{y}\right\|_2 \leq (1 + \epsilon) \left\|\mathbf{x} - \mathbf{y}\right\|_2 \quad \forall \mathbf{x}, \mathbf{y} \in P$$

is therefore called a Johnson–Lindenstrauss embedding of P. Note that while this property looks very similar to the definition of the restricted isometry property, it only holds for finite point sets. The JL lemma now asserts that the dimension m of the space has to be at least $m \gtrsim \log(N)$ for the above property to hold with high probability. In [8], this result was used in combination with a covering argument for the set of sparse vectors to provide an alternative RIP proof.

The statement of Theorem 11 depends on a yet unspecified constant C that effects the number of measurements required for a matrix to be an RIP matrix. For Gaussian matrices, the constant can be explicitly characterized (see [54, Chap. 9]). For example, in the asymptotic regime when $d/k \to \infty$, the RIP constant $\delta_{2k} \leq 0.6129$ can be achieved with probability at least $1 - \epsilon$ if

$$m \geq 54.868\left(k \log\left(\frac{ed}{2k}\right) + \frac{1}{2}\log(2\epsilon^{-1})\right). \tag{17}$$

Finally, it can be shown using tight bounds on the Gelfand widths of ℓ_1-balls that this bound on m is in fact optimal up to a constant [53, 67].

7.1.1 Bounded Orthonormal Systems

The random matrices discussed so far did not possess any discernible structure. However, in many domains of engineering, this assumption would be quite restrictive as the type of measurement matrix is often in part dictated by the specific application, be it due to the particular structure of the problem or for computational purposes. A typical example is structured random matrices involving the DFT or the Hadamard transform. In such situations, we may aim to exploit the existence of highly efficient numerical implementations such as fast Fourier transform (FFT) routines which might prevent us from incorporating a mixing stage involving random matrices into the acquisition system. Moreover, if fast implementations of the measurement operator are available, we can often exploit the operator in the decoding stage to drastically improve the efficiency of the employed recovery procedure. A canonical example of where structured random matrices emerge is when a band-limited function is to be constructed from random time-domain samples. In this case, we consider functions of the form

$$f(t) = \sum_{i=1}^{d} x_i \phi_i(t), \tag{18}$$

where $t \in \mathcal{D} \subset \mathbb{R}$ and the collection $\{\phi_i\}_i$ of functions from $\mathcal{D}$ to $\mathbb{C}$ forms a bounded orthonormal system (BOS) according to the following definition.[13]

Definition 14 (*Bounded orthonormal systems*) A collection of complex-valued functions $\{\phi_i\}_{i=1}^{d}$ defined on a set $\mathcal{D} \subset \mathbb{R}$ equipped with a probability measure μ is called a bounded orthonormal system with constant K if

$$\int_{\mathcal{D}} \phi_i(t)\phi_j(t)\mathrm{d}\mu(t) = \delta_{i,j}$$

and

$$\|\phi_i\|_\infty := \sup_{t \in \mathcal{D}} |\phi_i(t)| \leq K \ \forall i \in [d].$$

Let f be a function with a basis expansion as in Eq. (18) w. r. t. a bounded orthonormal system defined by the collection $\{\phi_i\}_i$. If we sample f at m points $t_1, \ldots, t_m \in \mathcal{D}$, we obtain the system of equations

$$y_j := f(t_j) = \sum_{i=1}^{d} x_i \phi_i(t_j), \quad j \in [m].$$

[13] The definition can easily be extended to the case where $\mathcal{D} \subset \mathbb{R}^n$, but we restrict our discussion to the scalar case here.

Collecting the samples $\left\{\phi_i(t_j)\right\}_j$ of the ith basis function in a vector $\boldsymbol{\phi}_i = (\phi_i(t_1), \ldots, \phi_i(t_m))^\top$ forming a column of the matrix $\mathbf{A} = \left[\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_d\right]$ of size $m \times d$, we immediately obtain the familiar form

$$\mathbf{y} = \mathbf{A}\mathbf{x},$$

where $\mathbf{y} = (y_1, \ldots, y_m)^\top$ and $\mathbf{x} = (x_1, \ldots, x_d)^\top$. As usual, we assume that $\mathbf{x}$ is sparse or compressible. In this case, the same recovery guarantees w. r. t. to the equality- or quadratically constrained basis pursuit problem can be established as soon as $\mathbf{A}$ or a scaled version of $\mathbf{A}$ can be shown to satisfy the restricted isometry property as before.

The reason why we endow $\mathcal{D}$ with a probability measure is of course that it allows us to draw the sampling points t_j from μ at random to establish the restricted isometry property of matrices defined w. r. t. subsampled bounded orthonormal systems probabilistically. Such results were first demonstrated in [28] for the case of the partial random Fourier matrix which satisfies the restricted isometry property with high probability provided we record $\boldsymbol{\Omega}(k \log^6(d))$ measurements. A nonuniform version of this result, which reduced the power of the log-term from 6 to 4, was shortly after proven by Rudelson and Vershynin in [89]. Another improvement was recently presented in [61] where the required number of measurements was further reduced to $\boldsymbol{\Omega}(k \log^2(k) \log(d))$ for randomly subsampled Fourier matrices. Under certain conditions, this bound can further be reduced. For instance, if the dimension d is an integer multiple of the sparsity level k, Bandeira et al. managed to remove the second log-factor in the previous bound, proving that $\boldsymbol{\Omega}(k \log(d))$ measurements suffice to establish the restricted isometry property for partial Fourier matrices [7]. In case the measurement matrix corresponds to a subsampled Hadamard matrix, Bourgain demonstrated in [17] the sufficiency of $\boldsymbol{\Omega}(k \log(k) \log^2(d))$ measurements to establish the restricted isometry property. A similar bound had previously been shown to hold by Nelson et al. in [74]. The best general bound to date asserts that $m = \boldsymbol{\Omega}(k \log^3(k) \log(d))$ measurements are required to establish the restricted isometry property for arbitrary subsampled bounded orthonormal systems where the sampling points are drawn from a discrete measure [32, Theorem 4.6]. This includes all measurement matrices formed by randomly selecting rows of a unitary matrix such as the DCT or DFT matrix, a Hadamard matrix, etc.

The following theorem records a modern general version of the RIP characterization for measurement matrices based on randomly subsampled bounded orthonormal systems.

Theorem 12 (BOS-RIP, [87, Theorem 4]) *Consider a set of complex-valued bounded orthonormal basis functions* $\left\{\phi_j\right\}_{j=1}^d$ *defined on a measure space* $\mathcal{D} \subset \mathbb{R}$ *equipped with the probability measure* μ*. Define a matrix* $\mathbf{A} \in \mathbb{C}^{m \times d}$ *with entries*

$$a_{ij} := \phi_j(t_i), \quad i \in [m], j \in [d],$$

constructed by independently drawing the sampling points t_i from the measure μ. Then with probability at least $1 - d^{-c\log^3(k)}$, the matrix $\frac{1}{\sqrt{m}}\mathbf{A}$ is an RIP matrix of order k with constant $\delta_k \leq \delta$ provided

$$m \geq C\delta^{-2}K^2 k \log^3(k)\log(d).$$

The positive constants C and c are universal.

For the existing bounds, the number of necessary measurements m scales with K^2. For the bound on m in Theorem 12 to be meaningful, the constant K should therefore either be independent of the dimension d or at least only scale with lower powers of d.

Finally, let us highlight that results as stated above can be extended to even more restrictive structured random matrices [6, 85]. For instance, the authors of [64] applied a novel technique to bound the suprema of chaos processes to obtain conditions under which random partial circulant matrices would satisfy the RIP. In this situation, the measurement procedure is of the form

$$\mathbf{Ax} = \frac{1}{\sqrt{m}}\mathbf{R}_{\boldsymbol{\Omega}}(\boldsymbol{\epsilon} * \mathbf{x}),$$

where $\mathbf{R}_{\boldsymbol{\Omega}} : \mathbb{C}^d \to \mathbb{C}^m$ denotes the operator restricting the entries of a vector to the set $\boldsymbol{\Omega} \subset [d]$ of cardinality m, $\boldsymbol{\epsilon}$ is a Rademacher vector of length d, and $*$ denotes the circular convolution operator. In general, if $m \geq C\delta^{-2}k\log^2(k)\log^2(d)$, then with probability at least $1 - d^{-\log(d)\log^2(k)}$ the partial random circulant matrix $\mathbf{A}$ satisfies the RIP of order k with constant $\delta_k \leq \delta$.

7.2 Random Matrices and the Null Space Property

While probabilistic constructions of RIP matrices have been established for a variety of random ensembles such as subgaussian distributions, as well as measurement matrices defined by randomly subsampled basis functions of bounded orthonormal systems as discussed in the previous section, there are some shortcomings to RIP-based recovery guarantees. For instance, the leading constants involved in the required scaling for Gaussian matrices to satisfy the RIP are often quite large. While these constants are usually due to artifacts of the proof strategy, analyses which establish stable and robust recovery by directly appealing to the null space property for Gaussian matrices often have much nicer constants. For instance, for large d and d/k with moderately large k, establishing the null space property requires $m \geq 8k\log(ed/k)$ measurements (cf. [54, Theorem 9.29]) which is much smaller than the constant involved in Eq. (17).

Another shortcoming in RIP-based analyses becomes evident when one tries to obtain recovery guarantees of the form

$$\left\| \mathring{\mathbf{x}} - \mathbf{x}^\star \right\|_q \leq C_{k,p}\sigma_k(\mathring{\mathbf{x}})_1 + D_k \left\| \mathbf{e} \right\|_p ,$$

where we aim to characterize the reconstruction performance in the presence of ℓ_p-bounded measurement noise for cases other than $(p, q) \in \{1, 2\}^2$. Note that we still measure the sparsity mismatch in terms of best k-term approximation error w. r. t. the ℓ_1-norm.[14] Such guarantees based on restricted isometries require a generalization of the restricted isometry property as stated in Definition 11. In particular, if one is interested in the recovery of a vector $\mathring{\mathbf{x}} \in \mathbb{C}^d$ from compressive measurements of the form $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}} + \mathbf{e}$ with $\left\| \mathbf{e} \right\|_p \leq \varepsilon$, we may solve the program

$$\begin{aligned} \text{minimize } & \left\| \mathbf{x} \right\|_1 \\ \text{s.t. } & \left\| \mathbf{A}\mathbf{x} - \mathbf{y} \right\|_p \leq \varepsilon. \end{aligned}$$

In order to characterize the reconstruction quality of a minimizer $\mathbf{x}^\star$ of this program, one may turn to the mixed (ℓ_p, ℓ_q)-RIP of the form

$$c \left\| \mathbf{x} \right\|_q \leq \left\| \mathbf{A}\mathbf{x} \right\|_p \leq C \left\| \mathbf{x} \right\|_q \quad \forall \mathbf{x} \in \Sigma_k.$$

However, as was recently addressed in [39], the best known probability bounds to establish the existence of such matrices for $p \neq 1, 2$ exhibit significantly worse scaling in the number of required measurements than $k \log(d/k)$. In their work, Dirksen et al. therefore derive concentration results which instead establish the ℓ_q-robust null space property (Definition 10), providing near-optimal scaling behavior of m (up to possible log-factors) [39] for more general heavy-tailed random matrices. In other words, they demonstrate that recovery guarantees as outlined above, which require similar scaling compared to the provably optimal regime in the case of the (ℓ_2, ℓ_2)-RIP, are not in general outside the realm of possibility. However, their work demonstrates that one may have to move away from RIP-type conditions, and consider stronger concepts such as the null space property and its generalizations to establish similar guarantees. Note that to the best of our knowledge, there currently do not exist any results which establish probabilistic bounds that directly assert the null space property of subsampled BOS matrices without first establishing the RIP to imply the null space property.

Finally, we want to point out two examples of measurement ensembles which provably require more than $k \log(d/k)$ measurements to satisfy the RIP but which nevertheless allow for typical recovery guarantees from $k \log(d/k)$ measurements. The first example is random matrices whose rows follow an isotropic log-concave distribution. Such matrices satisfy the canonical restricted isometry property, i.e., the (ℓ_2, ℓ_2)-RIP, only if $m \gtrsim k \log^2(ed/k)$ but provably allow for exact recovery as soon as $m \gtrsim k \log(ed/k)$ [2, 3, 63]. The second example concerns a certain combinatorial construction of sensing matrices based on the adjacency matrix of random left k-regular bipartite graphs with d left and m right vertices [12]. The corresponding

[14] This avoids another issue regarding the so-called instance optimality of pairs $(\mathbf{A}, \Delta)$ where $\Delta \colon \mathbb{C}^m \to \mathbb{C}^d$ denotes an arbitrary reconstruction algorithm (see [54, Chap. 11] for details).

graph is called a lossless expander and its normalized adjacency matrix $\frac{1}{s}\mathbf{A}$ can be shown to provide typical recovery guarantees with probability at least $1 - \eta$ if $s \gtrsim \log(ed/(k\eta))$ and $m \gtrsim k \log(ed/(k\eta))$. However, the matrix $\frac{1}{s}\mathbf{A}$ does not satisfy the (ℓ_2, ℓ_2)-RIP even though it satisfies the (ℓ_1, ℓ_1)-RIP.

8 An Algorithmic Primer

In the remainder of this introduction to compressed sensing, we want to turn our attention to the practical aspects of signal recovery. To that end, we decided to include a whirlwind tour of recovery algorithms that go beyond the scope of the quadratically constrained basis pursuit problem. Note, however, that the selection of algorithms chosen for this survey is not even close to exhaustive, and really only scratches the surface of what the literature holds in store. An informal search on the IEEE Xplore database produces upward of 1600 search results for the query “compressed sensing recovery algorithm.” Naturally, there is no doubt that this list includes a huge volume of work on specialized algorithms which go beyond the simple sparsity case that we will discuss in this section, as well as survey papers and works which simply benchmark the performance of existing algorithms in the context of specific problems. Nevertheless, this informal experiment still demonstrates the incredibly lively research activity in the field of recovery algorithms in compressed sensing and related domains. For that reason, we limit attention to only a handful of some of the most popular methods found in the pertinent literature and leave it up to the reader to inform him or herself beyond the methods surveyed in this section.

In general, there are multiple criteria by which authors have historically grouped different recovery algorithms for compressed sensing. The most generic classification usually considers three (mostly) distinct classes: convex optimization-based formulations,[15] so-called greedy methods, and iterative thresholding algorithms. Another possible classification could be based on the amount of prior knowledge required to run a particular algorithm. The most coarse classification in this regard takes the form of algorithms which require an explicit estimate of the sparsity level, and those which do not. As is the case for most other surveys on CS recovery algorithms, we decided to opt for the former here.

Before moving on to more efficient recovery methods (at least from a run time and computational complexity perspective), we first state some of the most common variants of convex problems one predominantly finds presented in the relevant literature.

[15] We are careful not to call this an algorithm class as optimization programs are technically just descriptions of problems which still require specialized algorithms such as interior-point methods to actually solve them.

8.1 Convex Programming

As usual, we model the measurement process of a perfectly sparse or compressible signal $\mathring{\mathbf{x}} \in \mathbb{C}^d$ via the affine model $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}} + \mathbf{e}$ where $\mathbf{e} \in \mathbb{C}^m$ is a norm-constrained noise term, i.e., $\|\mathbf{e}\|_p \le \eta$ with $\eta \ge 0$ and $p \ge 1$. If an upper bound, say w. r. t. the ℓ_2-norm, of this error term $\mathbf{e}$ is known, we naturally consider the quadratically constrained basis pursuit problem that we already discussed in Sect. 6.2:

$$\begin{aligned} &\text{minimize } \|\mathbf{x}\|_1 \\ &\quad\text{s.t.} \quad\ \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2 \le \eta. \end{aligned} \tag{QCBP}$$

For $\eta = 0$, this immediately reduces to the original basis pursuit problem.

Even though we already characterized the recovery behavior of this problem when we introduced the restricted isometry property, we state the result here again for completeness. If $\mathring{\mathbf{x}} \in \mathbb{C}^d$ is merely approximately sparse, one obtains the following characterization for minimizers $\mathbf{x}^\star$ of Problem (QCBP): if $\mathbf{A} \in \mathbb{C}^{m \times d}$ satisfies the restricted isometry property of order $2k$ with constant $\delta_{2k} < 4/\sqrt{41}$, one has [54, Theorem 6.12]

$$\left\|\mathbf{x}^\star - \mathring{\mathbf{x}}\right\|_2 \le C_1 k^{-1/2} \sigma_k(\mathring{\mathbf{x}})_1 + C_2 \eta, \tag{19}$$

where $C_1, C_2 > 0$ only depend on δ_{2k}. Clearly, this result implies perfect recovery in the case where we measure strictly k-sparse vectors in a noise-free environment.

For completeness, we also want to briefly highlight a few alternative convex programming formulations closely related to Problem (QCBP). A very common variant of the quadratically constrained basis pursuit program is the following unconstrained problem:

$$\text{minimize} \quad \|\mathbf{x}\|_1 + \lambda \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2 \tag{BPDN}$$

with $\lambda > 0$, often referred to as basis pursuit denoising (BPDN). The BPDN problem is particularly interesting in situations where no sensible estimate for the noise level η is available. In this case, one may instead use the parameter λ to control the trade-off between sparsity and data fidelity. Depending on the type of method used to solve this unconstrained problem, it might be helpful to replace the data penalty term $\|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2$ with its squared version to remove the differentiability issue. Of course, the nondifferentiability of the objective function of Problem (BPDN) remains unchanged by this step. However, if one employs a splitting-type algorithm where one alternates between optimizing over individual parts of the objective function, considering a squared ℓ_2-penalty enables us to use gradient-based techniques to deal with the smooth part of the problem. We will discuss an example of such an approach in Sect. 8.2.2 where we present a well-known iterative algorithm to solve a particular variation of Problem (BPDN).

Another important formulation is the so-called *least-absolute shrinkage selection operator* (LASSO) which was originally proposed in the context of sparse model selection in statistics:

$$\begin{aligned} \text{minimize } & \|\mathbf{Ax} - \mathbf{y}\|_2 \\ \text{s.t. } & \|\mathbf{x}\|_1 \leq \sigma. \end{aligned} \tag{LASSO}$$

Since the ℓ_1-norm generally functions as a sparsity prior, this formulation might be of interest in situations where rather than an estimate of the noise level η we might have access to a suitable estimate of the sparsity level. Recall that for $\mathring{\mathbf{x}} \in \Sigma_k$ we have by the Cauchy–Schwarz inequality that $\left\|\mathring{\mathbf{x}}\right\|_1 \leq \sqrt{k}\left\|\mathring{\mathbf{x}}\right\|_2$. Depending on the application of interest, an upper bound on the energy of the original signal $\mathring{\mathbf{x}}$ might be naturally available so that one may simply choose $\sigma = \sqrt{k}\left\|\mathring{\mathbf{x}}\right\|_2$.

Finally, the following program is known as the *Dantzig selector*:

$$\begin{aligned} \text{minimize } & \|\mathbf{x}\|_1 \\ \text{s.t. } & \|\mathbf{A}^*(\mathbf{Ax} - \mathbf{y})\|_\infty \leq \tau. \end{aligned} \tag{DS}$$

The key idea here is to impose a maximum tolerance on the worst-case correlation between the residuum $\mathbf{r} := \mathbf{Ax} - \mathbf{y}$ and the columns $\{\mathbf{a}_i\}_{i=1}^d$ of $\mathbf{A}$. In the extreme case $\tau = 0$, the Dantzig selector reduces to the classic basis pursuit problem since $\ker(\mathbf{A}^*) = \{\mathbf{0}\}$, and thus $\|\mathbf{A}^*(\mathbf{Ax} - \mathbf{y})\|_\infty = 0$ if and only if $\mathbf{x}$ belongs to the affine space $\left\{\mathbf{z} \in \mathbb{C}^d : \mathbf{Az} = \mathbf{y}\right\}$.

Conveniently, despite their different formulations and use cases, the problems (BPDN), (LASSO), and (DS) all share the same recovery guarantee from Eq. (19) up to nonlinear transformation of the parameters η, λ, and σ [54, Proposition 3.2]. While the Dantzig selector is the odd one out, similar guarantees can still be derived with relative ease. We refer interested readers to [20].

8.2 Thresholding Algorithms

While the recovery guarantees in the literature are usually strongest for convex optimization-based recovery procedures, generic solving algorithms based on interior-point methods [18, Chap. 11] as employed by popular optimization toolboxes like CVX [59] or CVXPY [38], as well as implementations more specialized to the particular nature of ℓ_1-minimization problems such as ℓ_1- MAGIC [19], SPGL1 [97] and YALL1 [105], become less and less practical if problem sizes increase. The class of thresholding algorithms represents an attractive compromise between strong theoretical guarantees and highly efficient and predictable running times.

Thresholding algorithms can generally be further subdivided into so-called *hard* and *soft-thresholding algorithms*. In the following, we present the most popular representatives from each class, namely, iterative hard thresholding (IHT) and hard thresholding pursuit (HTP) for the former, and the iterative soft-thresholding algorithm (ISTA) and the fast iterative soft-thresholding algorithm (FISTA) for the latter. Other popular thresholding-based algorithms include subspace pursuit [35], NESTA [10], and SpaRSA [103].

8.2.1 Hard Thresholding

At the heart of any hard thresholding algorithm lies the so-called *hard thresholding operator* $H_k \colon \mathbb{C}^d \to \Sigma_k$ defined as

$$H_k(\mathbf{x}) := \operatorname*{argmin}_{\mathbf{z} \in \Sigma_k} \|\mathbf{x} - \mathbf{z}\|_p,$$

for $p \geq 1$ which projects an arbitrary d-vector on the set of k-sparse vectors. The value $H_k(\mathbf{x})$ is constructed by identifying the index set $G \subset [d]$ of size $|G| = k$ which supports the largest values of $\mathbf{x}$ (in absolute value), and zeroing out any values supported on $\overline{G}$. In other words, the vector $H_k(\mathbf{x})$ achieves the best k-term approximation error $\sigma_k(\mathbf{x})_p$ for any $p \geq 1$. For convenience, we also define the set-valued operator $L_k \colon \mathbb{C}^d \to 2^{[d]}$ with $L_k := \operatorname{supp} \circ H_k$ yielding the support set of the best k-term approximation of $\mathbf{x} \in \mathbb{C}^d$. Here, 2^G denotes the power set of G.

With these definitions in place, we now turn to the first hard thresholding algorithm.

Iterative Hard Thresholding

The key idea of iterative hard thresholding is to reduce the smooth loss function $g(\mathbf{x}) := \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2$ with gradient $\nabla g(\mathbf{x}) = \mathbf{A}^*(\mathbf{A}\mathbf{x} - \mathbf{y})$ at every iteration by means of a gradient descent update before pruning the solution to the set of k-sparse vectors by means of the hard thresholding operator. The full listing of the algorithm is given in Algorithm 1.

Algorithm 1: Iterative Hard Thresholding (IHT)

```
Input: $\mathbf{A} \in \mathbb{C}^{m \times d}$, $\mathbf{y} \in \mathbb{C}^m$, $k \in [d]$
Initialization: $\mathbf{x}^0 \leftarrow \mathbf{0}$, $n \leftarrow 0$
while halting condition is not satisfied do
    $\mathbf{v}^{n+1} \leftarrow \mathbf{x}^n - \mathbf{A}^*(\mathbf{A}\mathbf{x}^n - \mathbf{y})$        Gradient descent step
    $\mathbf{x}^{n+1} \leftarrow H_k(\mathbf{v}^{n+1})$                              Projection on $\Sigma_k$
    $n \leftarrow n + 1$
end
Output: $\mathbf{x}^n$
```

Considering the nonlinearity of the operator H_k, it is not immediately obvious that Algorithm 1 even converges, let alone to the true solution $\mathring{\mathbf{x}}$. The following result demonstrates both robustness w. r. t. sparsity defect and stability w. r. t. measurement noise. Consider an arbitrary vector $\mathring{\mathbf{x}} \in \mathbb{C}^d$ which we measure according to the model $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}} + \mathbf{e}$. If $\mathbf{A}$ satisfies the RIP condition with constant $\delta_{6k} < 1/\sqrt{3}$, Algorithm 1 produces iterates $(\mathbf{x}^n)_{n \geq 0}$ satisfying [54, Theorem 6.21]

$$\left\|\mathbf{x}^n - \mathring{\mathbf{x}}\right\|_2 \leq 2\rho^n \left\|\mathring{\mathbf{x}}\right\|_2 + C_1 k^{-1/2} \sigma_k(\mathring{\mathbf{x}})_1 + C_2 \|\mathbf{e}\|_2 ,$$

where $C_1, C_2 > 0$, and $0 < \rho < 1$ are constants which only depend on δ_{6k}. For $n \to \infty$, this sequence converges to a cluster point $\mathbf{x}^\star$ satisfying

$$\left\|\mathbf{x}^\star - \mathring{\mathbf{x}}\right\|_2 \leq C_1 k^{-1/2} \sigma_k(\mathring{\mathbf{x}})_1 + C_2 \left\|\mathbf{e}\right\|_2 . \tag{20}$$

If the vector $\mathring{\mathbf{x}}$ we wish to recover is in reality supported on an index set $S \subset [d]$ of size k, and measurements are not disturbed by noise ($\mathbf{e} = \mathbf{0}$), one has $\sigma_k(\mathring{\mathbf{x}})_1 = 0$, and therefore $\left\|\mathbf{x}^\star - \mathring{\mathbf{x}}\right\|_2 \leq 0$, implying perfect recovery with $\mathbf{x}^\star = \mathring{\mathbf{x}}$.

Hard Thresholding Pursuit

The fundamental difference between IHT and HTP is the fact that HTP merely uses hard thresholded gradient descent updates to estimate the support set of $\mathring{\mathbf{x}}$. In particular, it propagates least-squares solutions of $\mathbf{y} = \mathbf{A}\mathbf{x}$ w. r. t. to a submatrix of $\mathbf{A}$ obtained by pursuing the active support set of coefficients in each iteration based on the operator $L_k = \operatorname{supp} \circ H_k$. A full algorithm listing is given in Algorithm 2.

Surprisingly, the stability and robustness analyses are identical for IHT and HTP barring a change of parameters (C_1, C_2, ρ) for HTP. Most importantly, this change results in a faster rate of convergence for the HTP algorithm [54].

Algorithm 2: Hard Thresholding Pursuit (HTP)

```
Input: $\mathbf{A} \in \mathbb{C}^{m \times d}$, $\mathbf{y} \in \mathbb{C}^m$, $k \in [d]$
Initialization: $\mathbf{x}^0 \leftarrow \mathbf{0}$, $n \leftarrow 0$
while halting condition is not satisfied do
    $\mathbf{v}^{n+1} \leftarrow \mathbf{x}^n - \mathbf{A}^*(\mathbf{A}\mathbf{x}^n - \mathbf{y})$        Gradient descent step
    $G_{n+1} \leftarrow L_k(\mathbf{v}^{n+1})$                                Support identification
    $\mathbf{x}^{n+1} \leftarrow \mathbf{0}$
    $\mathbf{x}^{n+1}_{G_{n+1}} \leftarrow \mathbf{A}^\dagger_{G_{n+1}}\mathbf{y}$                          Least-squares update
    $n \leftarrow n + 1$
end
Output: $\mathbf{x}^n$
```

8.2.2 Soft Thresholding

While the algorithms described in Sect. 8.2.1 rely on explicit hard thresholding to guarantee a certain sparsity level of solutions, soft-thresholding methods (also referred to as shrinkage thresholding for reasons which will become clear shortly) promote sparsity by incorporating an ℓ_1-prior in their objective functions, and applying the so-called *proximal gradient algorithm* or a variant thereof. In particular, we aim to solve the unconstrained regularized problem

$$\text{minimize} \;\; \lambda \left\|\mathbf{x}\right\|_1 + \frac{1}{2}\left\|\mathbf{A}\mathbf{x} - \mathbf{y}\right\|_2^2 , \tag{21}$$

with $\lambda > 0$. Up to rescaling of the objective function, and squaring of the ℓ_2-penalty, this is identical to Problem (BPDN) introduced earlier.

To explain the general idea behind soft thresholding, consider a loss function of the form $f(\mathbf{x}) := g(\mathbf{x}) + h(\mathbf{x})$ where $g\colon \mathbb{R}^d \cup \{-\infty, \infty\} \to \mathbb{R}$ is a (possibly) nonsmooth lower semi-continuous extended value function and $h\colon \mathbb{R}^d \to \mathbb{R}$ is a smooth convex function. If g were smooth, this problem could be solved by standard optimization tools such as (conjugate) gradient descent or Newton's method. However, in order to promote sparsity one will often choose $g = \lambda \|\cdot\|_1$, meaning that such a simple approach is not applicable. In the proximal gradient method, one therefore replaces the smooth part h of f by means of a second-order approximation, i.e., one considers an iterative approach of the form

$$\mathbf{x}^+ := \operatorname*{argmin}_{\mathbf{v}\in\mathbb{R}^d} \left\{ g(\mathbf{v}) + \hat{h}_t(\mathbf{x}, \mathbf{v}) \right\},$$

where $\mathbf{x}$ and $\mathbf{x}^+$ denote the current and next iterate, respectively, and

$$\hat{h}_t(\mathbf{x}, \mathbf{v}) := h(\mathbf{x}) + \langle \nabla h(\mathbf{x}), \mathbf{v} - \mathbf{x} \rangle + \frac{1}{2t} \|\mathbf{v} - \mathbf{x}\|_2^2 \tag{22}$$

with $t > 0$ is a second-order approximation of h around the point $\mathbf{x}$. It is easily verified that the expression for $\mathbf{x}^+$ can be rewritten as

$$\begin{aligned} \mathbf{x}^+ &= \operatorname*{argmin}_{\mathbf{v}\in\mathbb{R}^d} \left\{ g(\mathbf{v}) + h(\mathbf{x}) + \langle \nabla h(\mathbf{x}), \mathbf{v} - \mathbf{x} \rangle + \frac{1}{2t} \|\mathbf{v} - \mathbf{x}\|_2^2 \right\} \\ &= \operatorname*{argmin}_{\mathbf{v}\in\mathbb{R}^d} \left\{ g(\mathbf{v}) + \frac{1}{2t} \|\mathbf{v} - (\mathbf{x} - t\nabla h(\mathbf{x}))\|_2^2 \right\}. \end{aligned} \tag{23}$$

While this formulation might give the impression that we merely traded one difficult optimization problem for another, it turns out that the operator in Eq. (23) corresponds to the so-called *proximal operator* [77]

$$\operatorname{prox}_{tg}(\mathbf{x}) := \operatorname*{argmin}_{\mathbf{v}\in\mathbb{R}^d} \left\{ g(\mathbf{v}) + \frac{1}{2t} \|\mathbf{v} - \mathbf{x}\|_2^2 \right\},$$

applied to the gradient descent update $\mathbf{x} - t\nabla h(\mathbf{x})$. Conveniently, this operator has a closed-form solution for a variety of different nonsmooth functions g. In particular, it is easy to check via subdifferential calculus over its individual entries that $\operatorname{prox}_{\alpha\|\cdot\|_1}(\mathbf{x}) = S_\alpha(\mathbf{x})$ where

$$S_\alpha(x) := \begin{cases} \operatorname{sign}(x)(|x| - \alpha), & |x| \geq \alpha, \\ 0, & \text{otherwise}, \end{cases}$$

is the so-called shrinkage operator that is applied element-wise to $\mathbf{x}$.[16] Overall, we obtain the iteration

$$\mathbf{x}^{+} = S_{\lambda t}(\mathbf{x} - t\mathbf{A}^{\top}(\mathbf{A}\mathbf{x} - \mathbf{y})) \tag{24}$$

if we apply this method to the basis pursuit denoising Problem (21). In this particular formulation, the parameter t acts as a step size which we may choose (e.g.,) via backtracking line search, while $\lambda > 0$ can be used to control the trade-off between sparsity of the solution $\mathbf{x}^{\star}$ and the data fidelity term $\|\mathbf{A}\mathbf{x}^{\star} - \mathbf{y}\|_2$.

This algorithm requires on the order of $\mathcal{O}(1/\epsilon)$ iterations to come within an ϵ-range $|f(\mathring{\mathbf{x}}) - f(\mathbf{x}^n)| \leq \epsilon$ of optimality, implying a convergence rate of $\mathcal{O}(1/n)$ [9]. According to a celebrated result by Nesterov [75], the best achievable convergence rate in the class of nonsmooth first-order methods[17] is $\mathcal{O}(1/n^2)$. This rate is achievable by Nesterov's acceleration method, resulting in the well-known *fast iterative soft-thresholding algorithm* (FISTA) due to Beck and Teboulle when applied to the iterative soft-thresholding algorithm [9]. Informally, the key idea of FISTA is to add a momentum term depending on the last two iterates to avoid erratic changes in the search direction, i.e., one updates the iterates according to

$$\begin{aligned}\mathbf{v}^{n+1} &:= \mathbf{x}^n + \frac{n-2}{n+1}(\mathbf{x}^n - \mathbf{x}^{n-1}),\\ \mathbf{x}^{n+1} &:= S_{t_n}(\mathbf{v}^{n+1} - t_n\mathbf{A}^{\top}(\mathbf{A}\mathbf{x}^n - \mathbf{y}))\end{aligned}$$

with $t_n > 0$ the step size at iteration n. Note that this formulation, taken from [77], differs from the original one given in [9] which explicitly depends on the Lipschitz constant of the gradient of the smooth part of (21). Also note that while this algorithm obtains the desired convergence rate of $\mathcal{O}(1/n^2)$, it is not a descent method. In practice, this means that additional book keeping is required to keep track of the best current iterate. However, considering that this accelerated scheme virtually comes at the same computational cost as Eq. (24), the impact of book keeping is negligible if weighed against the greatly improved convergence behavior.

Both ISTA and FISTA solve the unconstrained problem (21), and provably converge to the global optimum at a linear and super-linear rate, respectively, where convergence without step size adaptation is determined by the Lipschitz constant $L := \|\mathbf{A}^{\top}\mathbf{A}\|_{2\to 2}$ of the gradient of $h(\mathbf{x}) := \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2$. Since our main objective is the recovery of sparse or more generally compressible vectors from noisy measurements, we still have to answer the question how closely these algorithms approximate the true solution $\mathring{\mathbf{x}}$, and under which conditions recovery is exact. Conveniently, these recovery guarantees can be expressed in terms of the guarantees obtained for the quadratically constrained basis pursuit problem stated in Sect. 8.1.

[16] Hence the name *shrinkage* thresholding.

[17] Note that while we used a second-order approximation of h in Eq. (22), we did so by approximating the Hessian $\nabla^2 h(\mathbf{x})$ as a scaled identity matrix, thereby ignoring the true second-order information of h.

This holds because—given a minimizer $\mathbf{x}^{\star}_{\text{QCBP}}$ of (QCBP)—we can always find a transformation $T(\mathbf{x}^{\star}_{\text{QCBP}}, \eta) = \lambda$ of the parameter $\eta \geq 0$ of (QCBP) and the parameter $\lambda > 0$ of the unconstrained problem (21) such that both convex problems have the same optimal value $f^{\star}$ [10]. Note, however, that explicitly finding the mapping T is generally a nontrivial problem [98].

It remains to show when Problem (21) has a unique minimizer such that the correspondence between the solutions $\mathbf{x}^{\star}_{\text{QCBP}}$ and $\mathbf{x}^{\star}_{\text{BPDN}}$ is one-to-one given an appropriate choice of parameters η and λ. To that end, one seeks conditions when minimizers of (21) are unique. While there are various publications that address the issue of uniqueness of solutions to this problem, e.g., [24, 94], none of them is immediately guaranteed by the RIP or NSP. For instance, [104, Theorem 4.1] establishes the following condition for minimizers of (21) to be unique.

Theorem 13 *Let* $\mathbf{x}^{\star}$ *be a minimizer of the basis pursuit denoising problem, and define* $S := \operatorname{supp}(\mathbf{x}^{\star})$. *Then* $\mathbf{x}^{\star}$ *is a unique minimizer iff*

1. $\mathbf{A}_S$ *has full column-rank,*
2. $\exists \mathbf{u} \in \mathbb{R}^m$ *such that* $\mathbf{A}_S^{\top}\mathbf{u} = \operatorname{sign}(\mathbf{x}^{\star}_S)$ *and* $\left\| \mathbf{A}_{\bar{S}}^{\top}\mathbf{y} \right\|_{\infty} < 1$.

Approximate Message Passing

Due to the structural similarity to the iterative soft-thresholding algorithm, we briefly touch upon another popular development in the field of iterative thresholding algorithms, namely, the so-called *approximate message passing* (AMP) method. Pioneered by Donoho et al. in [44], the general formulation of approximate message passing (AMP) closely resembles the basic form of ISTA. The difference amounts to a correction term of the residuum $\mathbf{r}^n = \mathbf{A}\mathbf{x}^n - \mathbf{y}$ stemming from the interpretation of the measurement model $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}}$ in terms of *loopy belief propagation* in graphical models. Based on a slight reformulation of Eq. (24), approximate message passing proceeds via the iterations

$$\mathbf{x}^{n+1} := S_{\mu_n}(\mathbf{A}^{\top}\mathbf{r}^n + \mathbf{x}^n), \tag{25}$$

$$\mathbf{r}^n := \mathbf{y} - \mathbf{A}\mathbf{x}^n + \frac{1}{\delta}\mathbf{r}^{n-1}\left\langle \mathbf{1}, S'_{\mu_n}(\mathbf{A}^{\top}\mathbf{r}^{n-1} + \mathbf{x}^{n-1})\right\rangle, \tag{26}$$

where $\delta := m/d$ and $S'_{\mu}(x)$ denotes the derivative of $S_{\mu}(x)$ ignoring the nondifferentiability at $|x| = \mu$. Despite this innocent looking correction term in Eq. (26) (also known as *Onsager correction*), which barely increases the computational complexity over ISTA, the performance of this algorithm in terms of the observed phase-transition diagrams turns out to be highly competitive with the de facto gold standard of ℓ_1-minimization and in certain situations even manages to outperform it [43].

The key ingredient to the success of AMP is the observation that in the large-system limit $m, d \to \infty$ with δ fixed, and $A_{ij} \sim_{\text{i.i.d.}} \mathsf{N}(0, 1/m)$, one has $\mathbf{A}^{\top}\mathbf{r}^n + \mathbf{x}^n = \mathring{\mathbf{x}} + \mathbf{v}^n$ for the argument of S_{μ_n} in Eq. (25) where $\mathbf{v}^n$ is an i.i.d. zero-mean Gaussian random vector whose variance σ_n^2—and hence the mean squared error (MSE) of the reconstruction—can be predicted by a state evolution formalism.

Since its original introduction, a variety of modifications and improvements have been proposed for the AMP algorithm. These include the denoising-based AMP (D-AMP) [69] which generalizes the state evolution formalism to general Lipschitz continuous denoisers other than the soft-thresholding function, vector AMP (V-AMP) [83] which extends AMP to more general classes of measurement matrices, and generalized AMP (GAMP) [82] which extends AMP to arbitrary input and output distributions and allows for dealing with nonlinearities in the measurement process. While the general versions of most of these AMP variants require some statistical knowledge about the parameters involved, there exist several modifications which estimate these parameters online via expectation maximization (EM).

In closing, we mention that Problem (21) can be tackled by a variety of related methods such as alternating direction method of multipliers (ADMM), forward–backward splitting, Douglas–Rachford splitting, or homotopy methods. We refer the interested reader to the excellent survey [50], as well as to the notes in [54, Chap. 15].

8.3 *Greedy Methods*

Greedy algorithms are generally characterized by their tendency to act according to locally optimal decision rules in hopes of eventually arriving at a global optimal solution. In particular, they never explicitly aim at minimizing a particular (non-)convex objective. Instead, they treat the collection of columns of the measurement matrix $\mathbf{A}$ as a dictionary of atoms $\{\mathbf{a}_i\}_{i=1}^d$ and first try to identify the atoms which likely contributed to the measurement vector $\mathbf{y}$, before estimating the associated weighting factors. Despite the fact that algorithms of this type had been in use long before the advent of compressed sensing, particularly in the image processing community, research into greedy algorithms for sparse recovery experienced a resurgence ever since the rise of compressed sensing. In this section, we will look at two of the most popular representatives in this particular class of algorithms, namely, the so-called orthogonal matching pursuit and compressive sampling matching pursuit methods.

Orthogonal Matching Pursuit

While technically a successor to the lesser used matching pursuit algorithm, orthogonal matching pursuit (OMP) remains to this day one of the most popular greedy algorithms due to the fact that it is one of the methods with the lowest footprint in terms of computational complexity. As can be seen from Algorithm 3, OMP updates its estimated support set one atom at a time by identifying the atom $\mathbf{a}_i$ that exhibits the strongest correlation with the residuum $\mathbf{r}^n = \mathbf{A}\mathbf{x}^n - \mathbf{y}$ as measured by the inner product $|\langle \mathbf{a}_i, \mathbf{r}^n \rangle|$.

The atom selection step in each OMP iteration can be interpreted as identifying the component of $\mathbf{x}^n$ w.r.t. which the function $f(\mathbf{x}^n) := \frac{1}{2}\|\mathbf{A}\mathbf{x}^n - \mathbf{y}\|_2^2$ varies the most. This is due to the fact that the gradient of f at $\mathbf{x}^n$ reads $\nabla(\frac{1}{2}\|\mathbf{A}\mathbf{x}^n - \mathbf{y}\|_2^2) = \mathbf{A}^*(\mathbf{A}\mathbf{x}^n - \mathbf{y}) = \mathbf{A}^*\mathbf{r}^n$. The update step $\mathbf{x}^n \to \mathbf{x}^{n+1}$ on the other hand corresponds

Algorithm 3: Orthogonal Matching Pursuit (OMP)

```
Input: $\mathbf{A} \in \mathbb{C}^{m \times d}$, $\mathbf{y} \in \mathbb{C}^m$, $k \in [d]$
Initialization: $\mathbf{x}^0 \leftarrow \mathbf{0}$, $G^0 \leftarrow \emptyset$, $n \leftarrow 0$, $\mathbf{r}^0 \leftarrow -\mathbf{A}^*\mathbf{y}$
while halting condition is not satisfied do
    $j_{n+1} \leftarrow \operatorname{argmin}_{j \in [d]} |(\mathbf{A}^*\mathbf{r}^n)_j|$        Atom identification
    $G_{n+1} \leftarrow G_n \cup \{j_{n+1}\}$        Support extension
    $\mathbf{x}^{n+1} \leftarrow \mathbf{A}^\dagger_{G_{n+1}}\mathbf{y}$        Least-squares projection
    $\mathbf{r}^{n+1} \leftarrow \mathbf{A}\mathbf{x}^{n+1} - \mathbf{y}$        Calculation of residuum
    $n \leftarrow n + 1$
end
Output: $\mathbf{x}^n$
```

to a projection of $\mathbf{y}$ on the subspace spanned by the columns of $\mathbf{A}$ indexed by the updated index set G_{n+1}.

While theoretical guarantees in the noise-free and exactly sparse case exist in abundance for OMP, robust and stable recovery guarantees are not as well-developed as one might expect given the maturity of the theory and the popularity of OMP in general. Oftentimes such results depend on additional regularity conditions on the class of vectors one aims to recover.

In general, OMP does not require an estimate of the sparsity level of the vector one aims to recover. The algorithm naturally terminates as soon as the same atom is selected twice in subsequent iterations. Other halting conditions include the relative change of estimates $\mathbf{x}^n$ between iterations and tolerance criteria of data fidelity measures w. r. t. $\mathbf{r}^n$. Considering that OMP updates the support set one index at a time per iteration, OMP requires at least k iterations to find a k-sparse candidate vector. If the sparsity level is known a priori, another natural termination condition is therefore simply given by the number of iterations.

One of the earliest recovery guarantees for OMP was the coherence-based condition $(2k-1)\mu < 1$ which allows OMP to recover any k-sparse vector from noiseless linear measurements in k iterations [42]. In light of the Welch bound (cf. Proposition 4)

$$\mu \geq \sqrt{\frac{d-m}{m(d-1)}},$$

this implies the quadratic scaling in the number of measurements announced in Sect. 6.3. Currently, one of the best known sufficiency conditions for exact k-sparse recovery in the noiseless setting in terms of the restricted isometry property requires $\delta_{k+1} < 1/\sqrt{k+1}$ [71, Theorem III.1].

In the general noise-corrupted setting with $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}} + \mathbf{e}$, one obtains the RIP-based bound [54, Theorem 6.25]

$$\left\|\mathbf{x}^{24k} - \mathring{\mathbf{x}}\right\|_2 \leq C_1 k^{-1/2} \sigma_k(\mathring{\mathbf{x}})_1 + C_2 \left\|\mathbf{e}\right\|_2 \tag{27}$$

for iterates of OMP after $24k$ iterations, where the constants $C_1, C_2 > 0$ only depend on the RIP constant $\delta_{26k} < 1/6$ of the associated measurement matrix $\mathbf{A}$. In the noiseless and exactly sparse case, Eq. (27) guarantees perfect recovery after $24k$ iterations. Note, however, that in this case OMP will already reach the global optimum after k iterations since the algorithm selects one atom per iteration, after which it will stall due to the fact that $\mathbf{r}^n = \mathbf{0}$ for $n > k$. Otherwise, the solution returned by OMP after $24k$ iterations could not be k-sparse.

These guarantees are a far cry from the recovery conditions one obtains for methods such as QCBP or IHT seeing how RIP matrices of order $26k$ are much harder to construct than matrices of order $2k$ and $3k$, respectively. One possible explanation for the demanding requirement on the RIP order of $\mathbf{A}$ is the fact that OMP in its presented form has no way to correct possibly erroneous choices of atoms made in previous iterations. In a sense, this observation can be seen as one of the main motivations of the compressive sampling matching pursuit algorithm we will introduce in the next section.

Compressive Sampling Matching Pursuit

The compressive sampling matching pursuit (CoSaMP) algorithm shares a lot of similarities both with the OMP algorithm and the hard thresholding pursuit algorithm described in Sect. 8.2. While technically also an iterative algorithm that relies on hard thresholding, it is usually considered an instance of the class of greedy algorithms. The full procedure is given in Algorithm 4. Given a current estimate $\mathbf{x}^n$ of $\mathring{\mathbf{x}}$,

Algorithm 4: Compressive Sampling Matching Pursuit (CoSaMP)

```
Input: $\mathbf{A} \in \mathbb{C}^{m \times d}, \mathbf{y} \in \mathbb{C}^m, k \in [d]$
Initialization $\mathbf{x}^0 \leftarrow \mathbf{0}, n \leftarrow 0, \mathbf{r}^0 \leftarrow -\mathbf{A}^*\mathbf{y}$
while halting condition is not satisfied do
    $G_{n+1} \leftarrow \mathrm{supp}(\mathbf{x}^n) \cup L_{2k}(\mathbf{A}^*\mathbf{r}^n)$          Support overestimation
    $\mathbf{v}^{n+1} \leftarrow \mathbf{0}$
    $\mathbf{v}^{n+1}_{G_{n+1}} \leftarrow \mathbf{A}^{\dagger}_{G_{n+1}}\mathbf{y}$          Least-squares projection
    $\mathbf{x}^{n+1} \leftarrow H_k(\mathbf{v}^{n+1})$          "Projection" on $\Sigma_k$
    $\mathbf{r}^{n+1} \leftarrow \mathbf{A}\mathbf{x}^{n+1} - \mathbf{y}$          Calculation of residuum
    $n \leftarrow n + 1$
end
Output: $\mathbf{x}^n$
```

CoSaMP proceeds by first identifying the $2k$ columns of $\mathbf{A}$ which best correlate with the residuum $\mathbf{r}^n = \mathbf{A}\mathbf{x}^n - \mathbf{y}$ at iteration n. The algorithm then continues to solve a least-squares problem w. r. t. to column submatrix defined by the support of $\mathbf{x}^n$ and the $2k$ column indices identified in the previous step. Since the algorithm ultimately aims to obtain strictly k-sparse solutions, the next estimate $\mathbf{x}^{n+1}$ is finally found via hard thresholding of the least-squares update $\mathbf{v}^{n+1}$.

Solving the least-squares problem over a column index set of size at most $3k$ effectively allows CoSaMP to adaptively correct previous choices of the support set of its estimate of $\mathring{\mathbf{x}}$. This is one of the main drawbacks of the OMP algorithm, which will never remove a previously selected atom $\mathbf{a}_i$ from its dictionary once column i of $\mathbf{A}$ was identified as an element contributing to $\mathbf{y}$.

In accordance with the previous algorithms, we once again state available stability and robustness results for CoSaMP. Consider a vector $\mathring{\mathbf{x}} \in \mathbb{C}^d$ which we aim to recover from its linear measurements $\mathbf{y} = \mathbf{A}\mathring{\mathbf{x}} + \mathbf{e}$ where $\mathbf{A} \in \mathbb{C}^{m\times d}$ satisfies the RIP of order $8k$ with $\delta_{8k} < 0.4782$. Then the sequence $(\mathbf{x}^n)_{n\geq 0}$ generated by Algorithm 4 satisfies [54, Theorem 6.28]

$$\left\|\mathbf{x}^n - \mathring{\mathbf{x}}\right\|_2 \leq 2\rho^n \left\|\mathring{\mathbf{x}}\right\|_2 + C_1 k^{-1/2}\sigma_k(\mathring{\mathbf{x}})_1 + C_2 \left\|\mathbf{e}\right\|_2, \tag{28}$$

where $C_1, C_2 > 0$, and $0 < \rho < 1$ only depend on δ_{8k}. Once again, Eq. (28) establishes the existence of cluster points $\mathbf{x}^\star$ satisfying

$$\left\|\mathbf{x}^\star - \mathring{\mathbf{x}}\right\|_2 \leq C_1 k^{-1/2}\sigma_k(\mathring{\mathbf{x}})_1 + C_2 \left\|\mathbf{e}\right\|_2,$$

which implies perfect recovery by convergence to the unique vector $\mathring{\mathbf{x}}$ once $\mathring{\mathbf{x}} \in \Sigma_k$ and $\mathbf{e} = \mathbf{0}$.

8.4 Iteratively Reweighted Least-Squares

Another popular method which does not quite fit into any of the categories discussed so far is the so-called iteratively reweighted least-squares (IRLS) algorithm. At its core, IRLS is motivated by the observation that

$$|x| = |x|^{-1}|x|^2$$

for $0 \neq x \in \mathbb{C}$. Assuming for the moment that $\mathring{\mathbf{x}} \in \Sigma_k$ were known, we could rewrite the basis pursuit problem as

$$\min\left\{\sum_{i=1}^{d} |x_i| : \mathbf{y} = \mathbf{A}\mathbf{x}\right\} = \min\left\{\sum_{i\in\operatorname{supp}(\mathring{\mathbf{x}})} |\mathring{x}_i|^{-1}|x_i|^2 : \mathbf{y} = \mathbf{A}\mathbf{x}\right\}. \tag{29}$$

The idea now is to treat the term $|\mathring{x}_i|^{-1}$ as a weighting factor that we iteratively update in an alternating fashion in between updates of the variables x_i. To that end, we define the weighting factors as a smooth approximation

$$w_i^{n+1} := |x_i^2 + \tau_{n+1}^2|^{-1/2}, \tag{30}$$

where we require $0 < \tau_{n+1} \leq \tau_n$ so that $w_i^{n+1} \to |x_i|^{-1}$ as $\tau_{n+1} \to 0$. Considering that $\operatorname{supp}(\mathring{\mathbf{x}})$ is unknown, this approximation has the added advantage that we can let the summation on the right-hand side of Eq. (29) run through all indices in $[d]$ as the regularization parameter τ_n avoids divisions by zero. To proceed, we now define the functional

$$\mathcal{F}(\mathbf{x}, \mathbf{w}, \tau) := \frac{1}{2}\left[\sum_{i=1}^{d} |x_i|^2 w_i + \sum_{i=1}^{d}(\tau^2 w_i + w_i^{-1})\right]. \tag{31}$$

This definition is motivated by the following observations. Given a fixed weight vector $\mathbf{w}$ and regularizer τ, Eq. (31) corresponds to Eq. (29) with $|\mathring{x}_i|^{-1}$ replaced by w_i. Defining $\mathbf{D}_\mathbf{w} := \operatorname{diag}\{\mathbf{w}\}$, this constitutes a least-squares minimization problem w. r. t. the induced norm $\|\mathbf{x}\|_{\mathbf{D}_\mathbf{w}} := \sqrt{\mathbf{x}^* \mathbf{D}_\mathbf{w} \mathbf{x}}$, i.e.,

$$\begin{aligned} &\text{minimize} \ \|\mathbf{x}\|_{\mathbf{D}_\mathbf{w}} \\ &\text{s.t.} \qquad \mathbf{y} = \mathbf{A}\mathbf{x}, \end{aligned}$$

which admits the closed-form solution

$$\mathbf{x}^\star = \mathbf{D}_\mathbf{w}^{-1/2}(\mathbf{A}\mathbf{D}_\mathbf{w}^{-1/2})^\dagger \mathbf{y}.$$

The second observation concerns the update of the weighting vector $\mathbf{w}$ given a fixed $\mathbf{x}$ and τ. In that case, it is easily verified for $i \in [d]$ that

$$w_i^\star = \operatorname*{argmin}_{w_i > 0} \mathcal{F}(\mathbf{x}, \mathbf{w}, \tau) = \frac{1}{\sqrt{|x_i|^2 + \tau^2}},$$

which corresponds to the regularization of w_i in terms of x_i and τ as motivated by Eq. (30). The full algorithm is listed in Algorithm 5. Note that the update rule for τ is chosen in such a way that τ_n is a nonincreasing sequence in n as motivated above.

The following recovery guarantee for the IRLS algorithm is based on [54, Theorem 15.15]. Let $\mathbf{A} \in \mathbb{C}^{m \times d}$ satisfy the restricted isometry property of order $2k$ with $\delta_{2k} < 7/(4\sqrt{41}) \approx 0.2733$, and define[18] for $\alpha_\delta := \sqrt{1 - \delta_{2k}^2} - \delta_{2k}/4$,

$$\rho := \frac{\delta_{2k}}{\alpha_\delta} \quad \text{and} \quad \tau := \frac{\sqrt{1 + \delta_{2k}}}{\alpha_\delta}.$$

Then the sequence $(\mathbf{x}^n)_{n \geq 0}$ generated by the IRLS algorithm converges to a point $\mathbf{x}^\star$, and

[18] Note that this choice amounts to $\mathbf{A}$ satisfying the ℓ_2-robust null space property (cf. Definition 10) of order k with constants $\rho < 1/3$ and $\tau > 0$ [54, Theorem 6.13].

$$\left\|\mathring{\mathbf{x}}-\mathbf{x}^{\star}\right\|_1 \leq \frac{2(3+\rho)}{1-3\rho}\sigma_k(\mathring{\mathbf{x}})_1$$

which implies perfect recovery via the IRLS algorithm if $\mathring{\mathbf{x}}$ is k-sparse.

Algorithm 5: Iteratively Reweighted Least-Squares (IRLS)

```
**Input**: $\mathbf{A} \in \mathbb{C}^{m \times d}$, $\mathbf{y} \in \mathbb{C}^m$, $k \in [d]$
**Initialization:** $\mathbf{w}^0 \leftarrow \mathbf{1}$, $n \leftarrow 0$, $\tau_0 \leftarrow 1$
**while** *halting condition is not satisfied* **do**
    $\mathbf{x}^{n+1} \leftarrow \mathbf{D}_{\mathbf{w}^n}^{-1/2}(\mathbf{A}\mathbf{D}_{\mathbf{w}^n}^{-1/2})^{\dagger}\mathbf{y}$
    $\tau_{n+1} \leftarrow \min\left\{\tau_n, (\mathbf{x}^n)^*_{k+1}/(2d)\right\}$
    $w_i^{n+1} \leftarrow \left(|x_i^{n+1}|^2 + \tau_{n+1}^2\right)^{-1/2} \quad \forall i \in [d]$
    $n \leftarrow n + 1$
**end**
**Output**: $\mathbf{x}^n$
```

9 Conclusion

In the years since its inception, the field of compressed sensing has steadily developed into a mature theory at the intersection of applied mathematics and engineering. With numerous applications in various domains of science and engineering, it now constitutes an indispensable tool in the toolbox of signal processing engineers who are faced with the problem of sampling high-dimensional signals in resource-constrained environments.

In this chapter, we reviewed some of the basic concepts of the theory, focusing on large part on the problem of nonuniform recovery of low-complexity signals from linear observations. In particular, we want to highlight the inclusion of a discussion on the connection between sparse recovery and conic integral geometry, a rather young development in the field, as well as a broader discussion of several efficient recovery algorithms and associated performance guarantees. We hope that the selection of topics featured in this introduction serves as a useful starting point in the further study of the theory of compressed sensing and its extensions.

Acknowledgements We would like to thank the anonymous reviewers and contributors to this book for their invaluable comments regarding this introduction.

References

1. S.I. Adalbjörnsson, A. Jakobsson, M.G. Christensen. Estimating multiple pitches using block sparsity, in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing* (May 2013), pp. 6220–6224
2. R. Adamczak, R. Latała, A.E. Litvak, A. Pajor, N. Tomczak-Jaegermann, Geometry of log-concave ensembles of random matrices and approximate reconstruction. C. R. Math. **349**(13), 783–786 (2011)
3. R. Adamczak, A.E. Litvak, A. Pajor, N. Tomczak-Jaegermann, Restricted isometry property of matrices with independent columns and neighborly polytopes by random sampling. Constr. Approx. **34**(1), 61–88 (2011)
4. D. Amelunxen, M. Lotz, M.B. McCoy, J.A. Tropp, Living on the edge: phase transitions in convex programs with random data. Inf. Inference **3**(3), 224–294 (2014)
5. U. Ayaz, S. Dirksen, H. Rauhut, Uniform recovery of fusion frame structured sparse signals. Appl. Comput. Harmon. Anal. **41**(2), 341–361 (2016)
6. W.U. Bajwa, J.D. Haupt, G.M. Raz, S.J. Wright, R.D. Nowak, Toeplitz-structured compressed sensing matrices, in *2007 IEEE/SP 14th Workshop on Statistical Signal Processing* (Aug. 2007), pp. 294–298
7. A.S. Bandeira, M.E. Lewis, D.G. Mixon, Discrete Uncertainty Principles and Sparse Signal Processing. J. Fourier Anal. Appl. **24**(4), 935–956 (2018)
8. R. Baraniuk, M. Davenport, R. DeVore, M. Wakin, A simple proof of the restricted isometry property for random matrices. Constr. Approx. **28**(3), 253–263 (2008)
9. A. Beck, M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J. Imaging Sci. **2**, 183–202 (2009)
10. S. Becker, J. Bobin, E.J. Candès, Nesta: A fast and accurate first-order method for sparse recovery. SIAM J. Imaging Sci. **4**, 1–39 (2011)
11. J. Bennett, S. Lanning, The netflix prize (2007)
12. R. Berinde, A.C. Gilbert, P. Indyk, H. Karloff, M.J. Strauss, Combining geometry and combinatorics: a unified approach to sparse signal recovery, in *2008 46th Annual Allerton Conference on Communication, Control, and Computing* (Sept. 2008), pp. 798–805
13. B.N. Bhaskar, G. Tang, B. Recht, Atomic norm denoising with applications to line spectral estimation. IEEE Trans. Signal Process. **61**(23), 5987–5999 (2011)
14. H. Boche, *Compressed Sensing and its Applications* (Springer Science+Business Media, New York, 2015)
15. P. Boufounos, G. Kutyniok, H. Rauhut, Sparse recovery from combined fusion frame measurements. IEEE Trans. Inf. Theory **57**(6), 3864–3876 (2011)
16. P.T. Boufounos, L. Jacques, F. Krahmer, R. Saab, Quantization and compressive sensing, in *Compressed Sensing and its Applications: MATHEON Workshop 2013*, Applied and Numerical Harmonic Analysis, ed. by H. Boche, R. Calderbank, G. Kutyniok, J. Vybíral (Springer International Publishing, Cham, 2015), pp. 193–237
17. J. Bourgain, An Improved Estimate in the Restricted Isometry Problem, in *Geometric Aspects of Functional Analysis*, vol. 2116, ed. by B. Klartag, E. Milman (Springer International Publishing, Cham, 2014), pp. 65–70
18. S. Boyd, L. Vandenberghe, *Convex Optimization* (Cambridge University Press, 2004)
19. E. Candes, J. Romberg, l1-magic: recovery of sparse signals via convex programming, vol. 4 (2005), p. 14. www.acm.caltech.edu/l1magic/downloads/l1magic.pdf
20. E. Candes, T. Tao, The Dantzig selector: statistical estimation when p is much larger than n. Ann. Stat. **35**(6), 2313–2351 (2007)
21. E.J. Candès, The restricted isometry property and its implications for compressed sensing. C. R. Math. **346**(9), 589–592 (2008)
22. E.J. Candes, D.L. Donoho, Curvelets-a surprisingly effective nonadaptive representation for objects with edges, in *Curves and Surfaces Fitting*, ed. by L.L. Schumaker, A. Cohen, C. Rabut (Vanderbilt University Press, Nashville, TN, 1999), p. 16

23. E.J. Candès, D.L. Donoho, New tight frames of curvelets and optimal representations of objects with piecewise c2 singularities. Commun. Pure Appl. Math. J. Issued Courant Inst. Math. Sci. **57**(2), 219–266 (2004)
24. E.J. Candes, Y. Plan, Near-ideal model selection by ℓ_1 minimization. Ann. Stat. **37**, 2145–2177 (2009)
25. E.J. Candès, J.K. Romberg, T. Tao, Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. IEEE Trans. Inf. Theory **52**, 489–509 (2006)
26. E.J. Candès, J.K. Romberg, T. Tao, Stable signal recovery from incomplete and inaccurate measurements. Commun. Pure Appl. Math. **59**(8), 1207–1223 (2006)
27. E.J. Candes, T. Tao, Decoding by linear programming. IEEE Trans. Inf. Theory **51**(12), 4203–4215 (2005)
28. E.J. Candès, T. Tao, Near-optimal signal recovery from random projections: universal encoding strategies? IEEE Trans. Inf. Theory **52**(12), 5406–5425 (2006)
29. A.Y. Carmi, L. Mihaylova, S.J. Godsill, *Compressed Sensing & Sparse Filtering* (Springer, 2016)
30. P.G. Casazza, G. Kutyniok, F. Philipp, Introduction to finite frame theory, in *Finite Frames* (Springer, 2013), pp. 1–53
31. V. Chandrasekaran, B. Recht, P.A. Parrilo, A.S. Willsky, The convex geometry of linear inverse problems. Found. Comput. Math. **12**(6), 805–849 (2012)
32. M. Cheraghchi, V. Guruswami, A. Velingker, Restricted isometry of Fourier matrices and list decodability of random linear codes. SIAM J. Comput. **42**(5), 1888–1914 (2013)
33. A. Cohen, W. Dahmen, R. Devore, Compressed sensing and best k-term approximation. J. Am. Math. Soc. 211–231 (2009)
34. R. Coifman, F. Geshwind, Y. Meyer, Noiselets. Appl. Comput. Harmon. Anal. **10**(1), 27–44 (2001)
35. W. Dai, O. Milenkovic, Subspace pursuit for compressive sensing signal reconstruction. IEEE Trans. Inf. Theory **55**, 2230–2249 (2009)
36. S. Dasgupta, A. Gupta, An elementary proof of a theorem of Johnson and Lindenstrauss. Random Struct. Algorithms **22**(1), 60–65 (2003)
37. R.A. DeVore, Nonlinear approximation. Acta Numer. **7**, 51–150 (1998)
38. S. Diamond, S. Boyd, Cvxpy: a python-embedded modeling language for convex optimization. J. Mach. Learn. Res. **17**(1), 2909–2913 (2016)
39. S. Dirksen, G. Lecué, H. Rauhut, On the gap between restricted isometry properties and sparse recovery conditions. IEEE Trans. Inf. Theory **64**(8), 5478–5487 (2018)
40. D.L. Donoho, Compressed sensing. IEEE Trans. Inf. Theory **52**, 1289–1306 (2006)
41. D.L. Donoho, M. Elad, Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ_1 minimization. Proc. Natl. Acad. Sci. **100**(5), 2197–2202 (2003)
42. D.L. Donoho, M. Elad, V.N. Temlyakov, Stable recovery of sparse overcomplete representations in the presence of noise. IEEE Trans. Inf. Theory **52**, 6–18 (2006)
43. D.L. Donoho, I. Johnstone, A. Montanari, Accurate prediction of phase transitions in compressed sensing via a connection to minimax denoising. IEEE Trans. Inf. Theory **59**, 3396–3433 (2013)
44. D.L. Donoho, A. Maleki, A. Montanari, Message passing algorithms for compressed sensing. Proc. Natl. Acad. Sci. U. S. A. **106**(45), 18914–9 (2009)
45. D.L. Donoho, J. Tanner, Observed universality of phase transitions in high-dimensional geometry, with implications for modern data analysis and signal processing. Philos. Trans. Ser. A Math. Phys. Eng. Sci. **367** (1906), 4273–4293 (2009)
46. M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. (Springer, New York, 2010). OCLC: ocn646114450
47. Y.C. Eldar, G. Kutyniok (eds.), *Compressed Sensing: Theory and Applications* (Cambridge University Press, Cambridge, 2012)
48. E. Elhamifar, R. Vidal, Sparse subspace clustering, in *2009 IEEE Conference on Computer Vision and Pattern Recognition* (June 2009), pp. 2790–2797

49. H.G. Feichtinger, T. Strohmer, *Gabor Analysis and Algorithms: Theory and Applications* (Springer Science & Business Media, 2012)
50. M. Fornasier, S. Peter, An overview on algorithms for sparse recovery, in *Sparse Reconstruction and Compressive Sensing in Remote Sensing*, ed. by X. Zhu, R. Bamler (Springer, June 2015), p. 76
51. M. Fornasier, H. Rauhut, Compressive sensing, in *Handbook of Mathematical Methods in Imaging*, ed. by O. Scherzer (Springer, New York, 2011), pp. 187–228. https://doi.org/10.1007/978-0-387-92920-0_6
52. S. Foucart, Flavors of compressive sensing, in *Approximation Theory XV: San Antonio 2016*, ed. by G.E. Fasshauer, L.L. Schumaker (Springer International Publishing, Cham, 2017), pp. 61–104
53. S. Foucart, A. Pajor, H. Rauhut, T. Ullrich, The Gelfand widths of ℓ_p-balls for $0<p \leq 1$. J. Complex. **26**(6), 629–640 (2010)
54. S. Foucart, H. Rauhut, *A Mathematical Introduction to Compressive Sensing* (Birkhäuser, Basel, 2013)
55. R. Foygel, L.W. Mackey, Corrupted sensing: novel guarantees for separating structured signals. IEEE Trans. Inf. Theory **60**, 1223–1247 (2014)
56. D. Goldberg, D. Nichols, B.M. Oki, D. Terry, Using collaborative filtering to weave an information tapestry. Commun. ACM **35**(12), 61–70 (1992)
57. Y. Gordon, On milman's inequality and random subspaces which escape through a mesh in $\mathbb{R}^n$, in *Geometric Aspects of Functional Analysis*, ed. by J. Lindenstrauss, V.D. Milman (Springer, Berlin, 1988), pp. 84–106
58. J. Gouveia, P.A. Parrilo, R.R. Thomas, Theta bodies for polynomial ideals. SIAM J. Optim. **20**, 2097–2118 (2010)
59. M. Grant, S. Boyd, Y. Ye, CVX: Matlab software for disciplined convex programming (2008)
60. Z. Han, H. Li, W. Yin, *Compressive Sensing for Wireless Networks* (Cambridge University Press, 2013)
61. I. Haviv, O. Regev, The restricted isometry property of subsampled fourier matrices, in *Geometric Aspects of Functional Analysis*, Lecture Notes in Mathematics (Springer, Cham, 2017), pp. 163–179
62. W.B. Johnson, J. Lindenstrauss, Extensions of lipschitz mappings into a hilbert space. Contemp. Math. **26**(189–206), 1 (1984)
63. V. Koltchinskii, *Oracle inequalities in empirical risk minimization and sparse recovery problems: École d'été de probabilités de Saint-Flour XXXVIII-2008*. Number 2033 in Lecture notes in mathematics. (Springer, Berlin, 2011). OCLC: ocn733246860
64. F. Krahmer, S. Mendelson, H. Rauhut, Suprema of chaos processes and the restricted isometry property. Commun. Pure Appl. Math. **67**(11), 1877–1904 (2014)
65. G. Kutyniok, D. Labate (eds.), *Shearlets: multiscale analysis for multivariate data*. Applied and Numerical Harmonic Analysis (Birkhäuser, New York, 2012). OCLC: ocn794844320
66. C. Liaw, A. Mehrabian, Y. Plan, R. Vershynin, A simple tool for bounding the deviation of random matrices on geometric sets (2016). CoRR, arXiv:1603.00897
67. G.G. Lorentz, M.V. Golitschek, Y. Makovoz, *Constructive Approximation: Advanced Problems* (Springer, Berlin, 2005). OCLC: 903339623
68. S.G. Mallat, *A Wavelet Tour of Signal Processing: The Sparse Way*, 3rd edn. (Elsevier/Academic Press, Amsterdam, 2009)
69. C.A. Metzler, A. Maleki, R.G. Baraniuk, From denoising to compressed sensing. IEEE Trans. Inf. Theory **62**, 5117–5144 (2016)
70. M. Mishali, Y.C. Eldar, Blind multiband signal reconstruction: compressed sensing for analog signals. IEEE Trans. Signal Process. **57**(3), 993–1009 (2009)
71. Q. Mo, A sharp restricted isometry constant bound of orthogonal matching pursuit (2015). CoRR, arXiv:1501.01708
72. B.K. Natarajan, Sparse approximate solutions to linear systems. SIAM J. Comput. **24**(2), 227–234 (1995)
73. S. Nathan, A. Shraibman, Rank, trace-norm and max-norm, in *COLT* (2005)

74. J. Nelson, E. Price, M. Wootters, New constructions of rip matrices with fast multiplication and fewer rows, in *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*. Society for Industrial and Applied Mathematics (2014), pp. 1515–1528
75. Y. Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course*, 1st edn. (Springer Publishing Company, Incorporated, 2014)
76. S. Oymak, B. Hassibi, New null space results and recovery thresholds for matrix rank minimization (Nov. 2010). arXiv:1011.6326 [cs, math, stat]
77. N. Parikh, S.P. Boyd, Proximal algorithms. Found. Trends Optim. **1**, 127–239 (2014)
78. F. Parvaresh, H. Vikalo, S. Misra, B. Hassibi, Recovering sparse signals using sparse measurement matrices in compressed dna microarrays. IEEE J. Sel. Top. Signal Process. **2**(3), 275–285 (2008)
79. Y. Plan, R. Vershynin, Robust 1-bit compressed sensing and sparse logistic regression: a convex programming approach. IEEE Trans. Inf. Theory **59**(1), 482–494 (2013)
80. Y. Plan, R. Vershynin, The generalized Lasso with non-linear observations. IEEE Trans. Inf. Theory **62**(3), 1528–1537 (2016)
81. Y.L. Polo, Y. Wang, A. Pandharipande, G. Leus, Compressive wide-band spectrum sensing, in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing* (Apr. 2009), pp. 2337–2340
82. S. Rangan, Generalized approximate message passing for estimation with random linear mixing, in*2011 IEEE International Symposium on Information Theory Proceedings* (2011), pp. 2168–2172
83. S. Rangan, P. Schniter, A.K. Fletcher, Vector approximate message passing, in *2017 IEEE International Symposium on Information Theory (ISIT)* (2017), pp. 1588–1592
84. N.S. Rao, B. Recht, R.D. Nowak, Universal measurement bounds for structured sparse signal recovery, in *AISTATS* (2012)
85. H. Rauhut, Circulant and Toeplitz matrices in compressed sensing, in *SPARS 09-Signal Processing with Adaptive Sparse Structured Representations* (Saint Malo, France, Apr. 2009), p. 7
86. H. Rauhut, K. Schnass, P. Vandergheynst, Compressed sensing and redundant dictionaries. IEEE Trans. Inf. Theory **54**(5), 2210–2219 (2008)
87. H. Rauhut, R. Ward, Sparse recovery for spherical harmonic expansions, in *Proceedings of the SampTA 2011* (2011)
88. R.T. Rockafellar, *Convex Analysis* (Princeton University Press, 2015)
89. M. Rudelson, R. Vershynin, On sparse reconstruction from Fourier and Gaussian measurements. Commun. Pure Appl. Math. **61**(8), 1025–1045 (2008)
90. S. Sarvotham, D. Baron, R.G. Baraniuk, Measurements vs. bits: compressed sensing meets information theory, in *Allerton Conference on Communication, Control and Computing* (2006)
91. M. Stojnic, ℓ_1 optimization and its various thresholds in compressed sensing, in *2010 IEEE International Conference on Acoustics, Speech and Signal Processing* (2010), pp. 3910–3913
92. G. Tang, B.N. Bhaskar, P. Shah, B. Recht, Compressed sensing off the grid. IEEE Trans. Inf. Theory **59**(11), 7465–7490 (2013)
93. R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, K. Knight, Sparsity and smoothness via the fused lasso. J. R. Stat. Soc. Ser. B (Stat. Methodol.) **67**(1), 91–108 (2005)
94. R.J. Tibshirani, The lasso problem and uniqueness (2012)
95. A.M. Tillmann, M.E. Pfetsch, The computational complexity of the restricted isometry property, the nullspace property, and related concepts in compressed sensing. IEEE Trans. Inf. Theory **60**, 1248–1259 (2014)
96. J.A. Tropp, Greed is good: algorithmic results for sparse approximation. IEEE Trans. Inf. Theory **50**(10), 2231–2242 (2004)
97. E. van den Berg, M.P. Friedlander, Spgl1: a solver for large-scale sparse reconstruction (2007)
98. E. van den Berg, M.P. Friedlander, Probing the pareto frontier for basis pursuit solutions. SIAM J. Sci. Comput. **31**(2), 890–912 (2008)
99. R. Vershynin, Introduction to the non-asymptotic analysis of random matrices, in *Compressed Sensing, Theory and Applications* (Cambridge University Press, Cambridge, 2012), pp. 210–268

100. R. Vershynin, *Estimation in High Dimensions: A Geometric Perspective* (Springer International Publishing, Cham, 2015), pp. 3–66

101. L. Welch, Lower bounds on the maximum cross correlation of signals (corresp.). IEEE Trans. Inf. Theory **20**(3), 397–399 (1974)

102. J. Wright, A.Y. Yang, A. Ganesh, S.S. Sastry, Y. Ma, Robust face recognition via sparse representation. IEEE Trans. Pattern Anal. Mach. Intell. **31**(2), 210–227 (2009)

103. S.J. Wright, R.D. Nowak, M.A.T. Figueiredo, Sparse reconstruction by separable approximation. IEEE Trans. Signal Process. **57**, 2479–2493 (2008)

104. H. Zhang, W. Yin, L. Cheng, Necessary and sufficient conditions of solution uniqueness in 1-norm minimization. J. Optim. Theory Appl. **164**, 109–122 (2015)

105. Y. Zhang, J. Yang, W. Yin, Yall1: your algorithms for l1 (2011). http://yall1.blogs.rice.edu

Quantized Compressed Sensing: A Survey

Sjoerd Dirksen

Abstract The field of quantized compressed sensing investigates how to jointly design a measurement matrix, quantizer, and reconstruction algorithm in order to accurately reconstruct low-complexity signals from a minimal number of measurements that are quantized to a finite number of bits. In this short survey, we give an overview of the state-of-the-art rigorous reconstruction results that have been obtained for three popular quantization models: one-bit quantization, uniform scalar quantization, and noise-shaping methods.

1 Introduction

In the last 15 years, compressed sensing [8, 9, 23, 29] has matured into a new paradigm in signal processing. This theory predicts that high-dimensional signals can be accurately reconstructed from a small number of measurements provided that the signal has low complexity. Whereas compressed sensing initially focused on the recovery of signals that can be approximately sparsely represented, many rigorous reconstruction results have been obtained for other low-complexity models, such as low-rank matrices and tensors, structured sparse signals, and signals located in a low-dimensional manifold, see e.g., [2, 15, 17, 24, 29, 50, 56] and the references therein.

In the standard compressed sensing model, one assumes that one has direct access to noisy analog linear measurements of the unknown signal x of the form $y = Ax + \nu$. In reality, these analog measurements need to be quantized to a finite number of bits before they can be transmitted, stored, and processed. This operation can be

S. Dirksen (✉)
Utrecht University, Mathematical Institute, P.O. Box 80010, 3508 Utrecht, TA, The Netherlands
e-mail: s.dirksen@uu.nl

H. Boche et al. (eds.), *Compressed Sensing and Its Applications*,
Applied and Numerical Harmonic Analysis,
https://doi.org/10.1007/978-3-319-73074-5_2

modeled by the application of a quantizer map $Q : \mathbb{R}^m \to \mathcal{Q}^m$, where $\mathcal{Q}$ is a finite (or sometimes, countable) alphabet. Accordingly, one has access to

$$q = Q(Ax + \nu). \tag{1}$$

Early works on compressed sensing assumed implicitly that the impact of quantization is negligible in the sense that the error due to the quantization step, i.e., $\eta = Q(Ax + \nu) - (Ax + \nu)$, is small in ℓ_2-norm, say. With this perspective, recovering x from (1) is simply a "usual" noisy compressed sensing problem and one can use standard methods, e.g., basis pursuit denoising, to recover the signal. This approach to recovery from quantized measurements, which we will call the *agnostic* approach, has two downsides. To ensure that the error η is small, one needs to use a very high-resolution quantizer, which may not be realistic or inefficient in practice, and even if this is possible, the estimates on the reconstruction error are pessimistic: the error will not decay beyond the noise floor, in particular not beyond the quantization error.

The area of *quantized compressed sensing* has shown that one can substantially improve over the agnostic approach by designing the triple $(A, Q, \mathscr{A})$ of measurement matrix A, quantizer Q and reconstruction algorithm $\mathscr{A}$ *in unison*. In the last few years, many fascinating results have been obtained in this area. The purpose of this survey is to give an introduction to the main emerging ideas. We do not intend to give an exhaustive overview of the area, but rather focus on rigorous reconstruction guarantees that have been obtained for three popular models in quantized compressed sensing: one-bit compressed sensing, uniform scalar quantization, and noise-shaping methods.

1.1 Notation

Throughout we will use the following notation. We reserve m for the number of measurements, n for the signal dimension, and ρ for the target reconstruction error. For any $N \in \mathbb{N}$ we write $[N] = \{1, \ldots, N\}$. We let $|S|$ denote the cardinality of a set S. We use $\|x\|_p$ to denote the ℓ_p-norm of a vector and $B_p^n = \{x \in \mathbb{R}^n : \|x\|_p \leq 1\}$. We write $\|x\|_0 = |\{i \in [n] : x_i \neq 0\}|$. We use S^{n-1} to denote the Euclidean unit sphere. d_H is the (unnormalized) Hamming distance on the discrete cube. For a random variable ξ we let $\|\xi\|_{L_p}$ denote its L_p-norm. We call ξ L-subgaussian if

$$\sup_{p \geq 1} \frac{\|\xi\|_{L^p}}{\sqrt{p}\|\xi\|_{L^2}} \leq L.$$

is finite. For a given measurement matrix $A \in \mathbb{R}^{m \times n}$ we let $a_1, \ldots, a_m$ denote its rows and refer to them as measurement vectors. We use $A^* \in \mathbb{R}^{n \times m}$ to denote the transpose of A. For a given $T \subset \mathbb{R}^n$ and $1 \leq p, q \leq \infty$, a matrix $A \in \mathbb{R}^{m \times n}$ is said to satisfy $\mathrm{RIP}_{p,q}(T, \varepsilon)$ if

$$(1-\varepsilon)\|x\|_q \leq \|Ax\|_p \leq (1+\varepsilon)\|x\|_q, \qquad \text{for all } x \in T. \tag{2}$$

We call a matrix $A \in \mathbb{R}^{m\times n}$ standard Gaussian if all its entries are i.i.d. standard Gaussian, Bernoulli if its entries are i.i.d. symmetric Bernoulli, or (L-)subgaussian if its entries are independent, mean-zero, unit variance, and (L-)subgaussian. For any $x \in \mathbb{R}^n$ we let $\Gamma_x \in \mathbb{R}^{n\times n}$ be the circulant matrix generated by x, i.e., $(\Gamma_x)_{i,j} = x_{(i-j) \bmod n}$. A circulant matrix implements the discrete circular convolution with x, i.e., $\Gamma_x z = x * z$ for all $z \in \mathbb{R}^n$. If ξ is a vector with independent, mean-zero, unit variance, (L-)subgaussian entries, then we call Γ_ξ an (L-)subgaussian circulant matrix. If the ξ_i are i.i.d. standard Gaussian or symmetric Bernoulli, then we call Γ_ξ a standard Gaussian or Bernoulli circulant matrix. A subsampled partial circulant matrix is obtained by selecting m rows from a circulant matrix. In the literature three different random selection models are considered, which we will give an explicit name here in order to distinguish between them. In the *row picking model*, one selects m rows independently of each other. Each row is picked uniformly at random from the set of $[n]$ rows of Γ_ξ. In the *uniformly at random model*, one selects a subset I uniformly at random from the set of all subsets of $[n]$ of cardinality m. One then considers the measurement matrix $R_I\Gamma_\xi$, where $R_I : \mathbb{R}^n \to \mathbb{R}^{|I|}$ is the operator defined by $R_I z = (z_i)_{i\in I}$. Finally, in the *selector model* one picks a vector $\theta \in \mathbb{R}^n$ of i.i.d. random selectors with mean m/n, sets $I = \{i \in [n] \, : \, \theta_i = 1\}$ and considers the measurement matrix $R_I\Gamma_\xi$. Note that $\mathbb{E}|I| = m$, so m corresponds to the expected number of measurements in this model.

If T is a closed set, then we let P_T be the ℓ_2-projection operator, which assigns to an element $x \in \mathbb{R}^n$ a certain solution of the optimization problem $\min_{z\in T} \|x - z\|_2$. In general, there is not a unique solution unless T is convex. For instance, if T is the set $\Sigma_s = \{x \in \mathbb{R}^n \, : \, \|x\|_0 \leq s\}$ of all s-sparse vectors, then $T = H_s$ is the hard thresholding operator. Finally, c and C denote absolute constants and their value many change from line to line. We use c_α or $c(\alpha)$ to denote a constant that only depends on the parameter α. We write $a \lesssim_\alpha b$ if $a \leq c_\alpha b$, and $a \simeq_\alpha b$ means that both $a \lesssim_\alpha b$ and $a \gtrsim_\alpha b$ hold.

2 Key Concepts

Before investigating the three different quantization models, we first introduce some important general concepts in quantized compressed sensing. We start by specifying the signals that we try to recover and the measurement matrices that we wish to analyze.

- **Low-complexity signal sets**. Any compressed sensing-type scheme exploits the fact that, even though the signal x that we would like to recover may be high-dimensional, it is a priori known to belong to a set of low *intrinsic dimension* or *complexity*. For instance, it is known empirically that many signals are (approximately) sparse in terms of a suitable basis, e.g., natural images can often be approximately sparsely represented in terms of wavelets. Accordingly, the number of measurements that need to be collected to ensure accurate reconstruction is governed by certain parameters that measure the complexity of the signal set. For our purposes, a suitable complexity measure is the *Gaussian width* of a bounded signal set $T \subset \mathbb{R}^n$, which is defined by

$$w(T) = \mathbb{E} \sup_{x \in T} \langle g, x \rangle,$$

where $g \in \mathbb{R}^n$ is standard Gaussian. Another measure that we will use is the ε-covering number $N(T, \varepsilon)$ of T, the minimal number of Euclidean balls of radius ε needed to cover T. The Gaussian width and covering numbers are closely related by Sudakov's and Dudley's inequality, which are the lower and upper bounds, respectively, in

$$\sup_{\varepsilon > 0} \varepsilon \sqrt{\log N(T, \varepsilon)} \lesssim w(T) \lesssim \int_0^\infty \sqrt{\log N(T, \varepsilon)}\, d\varepsilon.$$

Neither of the two bounds is sharp in general, see e.g., [62] for more details. Several of the results that we discuss below state rigorous reconstruction guarantees for a general signal set T and give a bound on the sufficient number of measurements for recovery in terms of the Gaussian width and covering numbers. Other results only concern sparse recovery. To allow for easy comparison, let us recall the following. If $\Sigma_s = \{x \in \mathbb{R}^n \ : \ \|x\|_0 \leq s\}$ is the set of sparse signals, then $w^2(\Sigma_s \cap B_2^n) \simeq s \log(en/s)$ and $\log N(\Sigma_s \cap B_2^n, \rho) \lesssim s \log(en/(s\rho))$. As a model for approximate sparsity, we also consider the larger set of *s-effectively sparse* signals $\Sigma_s^{\text{eff}} = \{x \in \mathbb{R}^n \ : \ \|x\|_1 \leq \sqrt{s}\|x\|_2\}$. If x is s-effectively sparse and $\|x\|_2 \leq 1$, then x belongs to the set of *s-compressible* signals $\sqrt{s}B_1^n \cap B_2^n$. The latter set is essentially the convex hull of the set of s-sparse vectors in the unit ball (see [53, Lemma 3.1]):

$$\operatorname{conv}(\Sigma_s \cap B_2^n) \subset \sqrt{s}B_1^n \cap B_2^n \subset 2\operatorname{conv}(\Sigma_s \cap B_2^n). \tag{3}$$

Since the Gaussian width is invariant under taking convex hulls, one finds $w^2(\sqrt{s}B_1^n$
$\cap B_2^n) \simeq s \log(en/s)$.

- **Random matrices**. Similarly to the situation in "unquantized" compressed sensing, the best-known recovery guarantees in quantized compressed sensing have been obtained for *random* measurement matrices. In particular, in quantized compressed sensing, optimal results have been obtained for standard Gaussian measurement matrices, i.e., matrices with independent standard Gaussian entries. These results are mostly of theoretical interest, as these matrices are difficult to realize in a practical measurement setup. On the other hand, it has proven very challenging to establish recovery guarantees for deterministic measurement matrices involving a number of measurements that is close to optimal. As a compromise between completely random matrices and deterministic ones, it is of interest to study *structured random matrices*, which arise when introducing randomness in (more) realistic measurement models. Two particularly popular classes of matrices, which can be considered as the "fruitflies" of compressed sensing with structured matrices, are *partial random circulant matrices* and *randomly subsampled bounded orthonormal systems*. The former model is connected to SAR radar imaging, Fourier optical imaging, and channel estimation (see e.g., [58] and the references therein). The latter model is relevant to many applications, for instance, models in compressive magnetic resonance imaging [47]. In standard compressed sensing it has been shown that stable and robust sparse recovery can be achieved with a near-optimal (i.e., up to logarithmic factors) number of measurements, see [7, 35, 44, 49, 59] for the best known bounds for the two respective classes of matrices. Recently, substantial progress has been made on quantized compressed sensing with structured random matrices. We will mostly restrict our discussion to (sub)gaussian matrices and circulant matrices, as results for these matrices have been obtained for all three quantization models that we consider in this survey.

Let us now discuss some terminology regarding quantization.

- **Memoryless versus adaptive schemes**. The quantizer $Q : \mathbb{R}^m \to \mathcal{Q}^m$ is called *memoryless* if it quantizes each entry of its input vector independently of the others. In contrast, an *adaptive* quantizer quantizes the i-th measurement using knowledge of previous analog measurements, their quantizations, and in some cases, even reconstructions of the signal based on the previous $i-1$ quantized measurements. As we will discuss below, adaptive methods can achieve a fundamentally better error decay rate. Whereas the reconstruction error cannot decay faster than linear (i.e., as $O(1/m)$) in terms of the number of measurements if a memoryless scalar quantization scheme is used, adaptive schemes can achieve a polynomial or even an optimal exponential error decay rate. This improved rate comes at a price: the implementation of adaptive schemes generally requires hardware that is more complicated and consumes more energy in operation. In addition, since by their very nature adaptive methods require measurements to be acquired sequentially, their implementation may be difficult or impossible in some sensing scenarios, e.g., in distributed sensing with a sensor network.
- **Dithering**. In the engineering literature on quantization, it has been known for a long time (at least since the work [57], see also [31, 32]) that it is potentially helpful to add random noise to the analog measurements before quantizing. This operation is called *dithering*. Note that the term "random noise" is somewhat misleading, since at least we have the freedom to *design* the distribution of the dithering vector. Indeed, as we will see below, it was recently shown rigorously that dithering with *well-chosen* distributions can substantially improve reconstruction guarantees in quantized compressed sensing.

Finally, we formalize some concepts regarding recovery methods.

- **Uniform versus non-uniform recovery**. The reconstruction results in quantized compressed sensing involving random matrices or dithering are guarantees to reconstruct a signal x or a class of signals with "high probability", which typically means that recovery will only fail with a probability that decays exponentially in terms of the number of measurements. These results can either be *uniform*, meaning that a high probability event exists upon which one can reconstruct any signal $x \in T$ (e.g., the set of all sparse vectors with unit norm), or *non-uniform*, meaning that the high probability event depends on the specific signal x which is to be recovered. Accordingly, a uniform guarantee is sometimes informally called a "for all" guarantee, whereas a non-uniform one is called a "for one" guarantee. To understand the difference between the two from a practical point of view, suppose that $A = R_I U$ is a randomly subsampled unitary matrix and suppose that T is the set of all s-sparse vectors on the unit sphere. A uniform recovery guarantee means that when we draw a random sample of the rows of U then, with high probability, we can recover any unit norm s-sparse vector from $Q(Ax + \nu)$. Thus, with high probability, a *single* random draw of the rows will yield a matrix that can be used for compressed sensing of *any* signal from the set T. A non-uniform guarantee is much weaker: only for a *fixed* signal x one shows that with high probability one can draw a random subset of the rows so that x can be recovered from its measurements. Hence, in this setting, we only guarantee good reconstruction performance with high probability if we draw a new random subset of the rows of U each time that we measure a new signal.
- **Quantization consistency**. A vector $x^{\#}$ is called *quantization consistent* with the true signal x if, when we were to measure and quantize $x^{\#}$, we would reproduce the observed quantized measurements. For instance, if we observe $q = Q(Ax)$, then $x^{\#}$ is quantization consistent if $q = Q(Ax^{\#})$. Several successful reconstruction methods that will be introduced below search for a quantization consistent vector.
- **Stability and robustness**. A triple $(A, Q, \mathscr{A})$ can only be expected to perform satisfactorily if it is *stable* and *robust*. We say that it is stable if the reconstruction performance does not deteriorate sharply if the signal lies "slightly outside of" the low-complexity set T. For instance, in the context of sparse recovery it is desirable to be able to accurately recover vectors that are not exactly sparse, but only effectively sparse or compressible. In addition, we would like to ensure that $(A, Q, \mathscr{A})$ is robust with respect to both *pre-quantization noise*, i.e., the noise ν on the analog measurements, as well as *post-quantization noise*, i.e., bit corruptions occurring during the quantization process.

3 Two Fundamental Limits

To set benchmarks for the reconstruction results for the three different quantization models, let us first formulate two fundamental lower bounds for the recovery error. The first concerns a lower bound for (uniform) recovery of signals from a set T in terms of its covering numbers. Suppose that we wish to quantify how many bits we need to collect to ensure that the worst case ℓ_2-reconstruction error of a reconstruction map $\mathscr{A}$ over the set T, i.e.,

$$\sup_{x\in T} \|x - \mathscr{A}(Q(Ax+\nu))\|_2,$$

is at most ρ. If this is fulfilled, then the set of Euclidean balls with radii ρ and centers in the image set $\mathscr{A}(Q(Ax+\nu))$ form a covering of T. If our quantization scheme Q encodes any analog measurement vector $Ax+\nu$ into at most B bits, then this cover has at most 2^B elements. Thus, the minimal total number of bits required to attain worst case error ρ over T satisfies

$$B \geq \log_2 N(T,\rho).$$

In particular, if we collect L bits per measurement, then at least $m \gtrsim \log_2 N(T,\rho)/L$ measurements are necessary. As an example, $\log_2 N(T,\rho) \simeq s\log_2(1/\rho)$ if T is the intersection of the Euclidean unit sphere with an s-dimensional subspace, so the worst case reconstruction error cannot decay faster than exponential in terms of the number of measurements in this case. In particular, one cannot obtain a better worst case error decay rate for the set of s-sparse vectors on the sphere.

The second fundamental lower bound concerns non-uniform recovery of sparse vectors.

Theorem 1 ([21, Theorem 1.3]) *Suppose that ν contains i.i.d. centered Gaussian random variables with variance σ^2. Let A be a (random) measurement matrix that satisfies, with probability at least* 0.95*,*

$$\|Ax\|_2 \leq \kappa\sqrt{m}\|x\|_2, \qquad \text{for all } x \in \Sigma_s \cap B_2^n. \tag{4}$$

Let Ψ be any recovery procedure such that, for every fixed $x \in \Sigma_s \cap B_2^n$, when receiving as data the measurement matrix A and the noisy linear measurements $Ax+\nu$, Ψ returns $x^\sharp$ that satisfies $\|x^\sharp - x\|_2 \leq \rho$ with probability 0.9*. Then*

$$m \geq c\kappa^{-2}\sigma^2 \frac{s\log(en/s)}{\rho^2}.$$

Note that the condition (4) is satisfied by many popular random measurement matrices if $m \gtrsim s\log^\alpha(n)$, in particular by subgaussian matrices, partial subgaussian circulant matrices and randomly subsampled bounded orthonormal systems. For these matrices the sample size required for recovery with accuracy ρ is at least $\sigma^2 s\log(en/s)/\rho^2$, even if one receives the noisy analog linear measurements *prior to quantization*, and is then free to use those measurements as one sees fit. In particular, in a high noise setting one cannot hope to achieve a better error decay rate than $O(1/\sqrt{m})$.

4 One-Bit Compressed Sensing

We start by discussing *one-bit compressed sensing*, which studies the extreme case where each measurement is quantized to a *single* bit. Specifically, we consider the map $Q_\tau : \mathbb{R}^m \to \{-1, 1\}^m$ defined by $Q_\tau(z) = \mathrm{sign}(z + \tau)$, where sign is the signum function applied element-wise and $\tau \in \mathbb{R}^m$ is a vector of quantization thresholds. This quantizer is memoryless if τ is a fixed or a randomly generated vector. In this case, the one-bit quantizer can be easily implemented by voltage comparison to fixed thresholds (τ deterministic) combined with dithering (τ random). Due to the efficiency of the memoryless one-bit quantizer, one-bit compressed sensing is one of the most popular quantized compressed sensing models. For a memoryless one-bit quantizer we cannot expect better than linear decay of the reconstruction error [6, 30, 42]. However, as we will see in Sect. 4.4, optimal error decay can be achieved by choosing the thresholds adaptively.

In the context of one-bit compressed sensing, post-quantization noise takes the form of "bit flips": the quantizer erroneously produces the bit $-q_i$ rather than $q_i = \mathrm{sign}(\langle a_i, x\rangle + \tau_i)$. One can either assume that bit corruptions occur in a random fashion, i.e., one observes a vector $q_c \in \{-1, 1\}^m$ satisfying $(q_c)_i = f_i q_i$, where the f_i are independent random variables satisfying $\mathbb{P}(f_i = -1) = 1 - \mathbb{P}(f_i = 1) = p$, i.e., a bit is corrupted with probability p. Alternatively, one can assume that a small fraction β of the bits are arbitrarily corrupted, i.e., one observes a vector $q_c \in \{-1, 1\}^m$ satisfying $d_H(q, q_c) \le \beta m$. Clearly, the second noise model is more challenging to analyze, as bit corruptions can in principle occur in an adversarial fashion.

4.1 Memoryless One-Bit Compressed Sensing: Zero Thresholds

One-bit compressed sensing was first considered by Boufounos and Baraniuk [5] in the completely noiseless case (i.e., neither pre- nor post-quantization noise) and $\tau = 0$. In this case, one simply observes $q = \mathrm{sign}(Ax)$. Since the sign function is invariant under positive scaling, the energy $\|x\|_2$ of the signal x is lost during quantization and one can only hope to recover its direction $x/\|x\|_2$. For this reason, it is standard in this original one-bit compressed sensing model to assume that $\|x\|_2 = 1$. From a geometric perspective, the vector q is a rough encoding of the position of x on S^{n-1}. To see this, note that each measurement vector a_i (i.e., the i-th row of A) determines a hyperplane $H_{a_i} = \{z \in \mathbb{R}^n \ : \ \langle a_i, z\rangle = 0\}$ passing through the origin. The corresponding quantized measurement $\mathrm{sign}(\langle a_i, x\rangle)$ indicates on which side of the hyperplane x is located. By taking m measurements, the space $\mathbb{R}^n$ is tessellated into (at most) 2^m cells, and the bit sequence $q = \mathrm{sign}(Ax) = (\mathrm{sign}(\langle a_i, x\rangle))_{i=1}^m \in \{-1, 1\}^m$ encodes in which cell x is located.

The original paper [5] considered recovery of a sparse vector from its one-bit measurements and proposed to reconstruct the signal via

$$\min_{z\in\mathbb{R}^n} \|z\|_0 \qquad \text{s.t.} \qquad q = \operatorname{sign}(Az), \ \|z\|_2 = 1. \tag{5}$$

The linear constraint $q = \operatorname{sign}(Az)$ forces any solution $x^\#$ to (5) to be quantization consistent. Geometrically, a vector z is quantization consistent with x precisely when it is located in the same cell of the hyperplane tessellation induced by the quantized measurements. To show that one can recover any $x \in \Sigma_s \cap S^{n-1}$ via (5) up to error ρ, one therefore needs to ensure that the measurement vectors tessellate $\Sigma_s \cap S^{n-1}$ into cells with diameter at most ρ. It was shown in [42, Theorem 2] that standard Gaussian vectors have this property: if $A \in \mathbb{R}^{m\times n}$ is standard Gaussian and $m \gtrsim \rho^{-1} s \log(n/\rho)$ then, with high probability, any s-sparse x, x' with $\|x\|_2 = \|x'\|_2 = 1$ and $\operatorname{sign}(Ax) = \operatorname{sign}(Ax')$ satisfy $\|x - x'\|_2 \le \rho$. In particular, any solution $x^\#$ to (5) satisfies $\|x^\# - x\|_2 \le \rho$. The number of measurements needed for this reconstruction is essentially optimal: in fact, the reconstruction $x^\#$ of an s-sparse vector produced by *any* method using $\operatorname{sign}(Ax)$ as its input must satisfy the lower bound $\|x^\# - x\|_2 \gtrsim s/(m + s^{3/2})$ [42, Theorem 1]. Hence, the reconstruction error cannot decay faster than linear (i.e., than $O(1/m)$). This linear decay bottleneck is common to all memoryless scalar quantization methods, see Sect. 5.

Even though the error of the reconstruction produced by (5) decays essentially optimally if A is standard Gaussian, this program is hard to solve. Although one can convexify the objective of (5) by replacing $\|z\|_0$ by $\|z\|_1$, the constraint $\|z\|_2 = 1$ is problematic (note that the relaxation $\|z\|_2 \le 1$ leads to a trivial program). A solution to this problem was proposed by Plan and Vershynin [53]: the simple, yet effective, idea is to observe that if A is standard Gaussian, then for any $z \in \mathbb{R}^n$,

$$\frac{1}{m}\mathbb{E}\|Az\|_1 = \sqrt{\frac{2}{\pi}}\|z\|_2.$$

This suggests to use the reconstruction program

$$\min_{z\in\mathbb{R}^n} \|z\|_1 \qquad \text{s.t.} \qquad q = \operatorname{sign}(Az), \ \|Az\|_1 = m\sqrt{\frac{2}{\pi}}, \tag{6}$$

which is a linear program. Plan and Vershynin showed that using $m \gtrsim \rho^{-5} s \log^2(n/s)$ standard Gaussian measurements one can, with high probability, recover every $x \in \mathbb{R}^n$ with $\|x\|_1 \le \sqrt{s}$ and $\|x\|_2 = 1$ via (6) up to reconstruction error ρ. This was the first uniform reconstruction result for stable recovery of sparse vectors from their one-bit measurements via a tractable program. Still, the program (6) has a weakness, which is common to any recovery program that enforces quantization consistency: the program can easily fail in the presence of post-quantization noise. Indeed, already a single bit corruption can cause (6) to be infeasible: there will simply be no vector

z which is consistent with the observed corrupted quantized measurements (see [20] for a detailed discussion).

In order to handle post-quantization noise, Plan and Vershynin introduced a different program in [54], which can be used to robustly reconstruct signals from an arbitrary set $T \subset S^{n-1}$, namely

$$\max_{z\in\mathbb{R}^n}\langle q_c, Az\rangle \qquad \text{s.t.} \qquad z \in T. \tag{7}$$

That is, we search for a vector that maximizes the correlation between the linear and observed corrupted quantized measurements. This program is convex if T is convex and therefore [54] suggested to use this program with $T = \operatorname{conv}(\Sigma_s \cap B_2^n)$ for stable sparse recovery. By (3), this leads to the tractable program

$$\max_{z\in\mathbb{R}^n}\langle q_c, Az\rangle \qquad \text{s.t.} \qquad \|z\|_1 \leq \sqrt{s}, \;\; \|z\|_2 \leq 1.$$

In a non-uniform recovery setting, Plan and Vershynin showed that $m \gtrsim \rho^{-4} w^2(T)$ measurements suffice to reconstruct a fixed signal in T with high probability up to error ρ, even if pre-quantization noise is present and quantization bits are randomly flipped with a probability that is allowed to be arbitrarily close to $1/2$. A much deeper result is the following uniform recovery theorem, which proves robustness of (7) to adversarial post-quantization noise.

Theorem 2 ([54, Theorem 1.3]) *Fix $0 < \rho, \beta \leq 1$, let $T \subset B_2^n$ and let $A \in \mathbb{R}^{m\times n}$ be standard Gaussian. Suppose that*

$$m \geq c_2 \frac{\log^3(e/\rho)}{\rho^{12}} w^2(T), \qquad \beta\sqrt{\log(e/\beta)} = c_3\rho^2.$$

Then with probability at least $1 - e^{-c_1 m\rho^4/\log(e/\rho)}$ the following holds for any $x \in T$ with $\|x\|_2 = 1$. If we observe $q_c \in \{-1, 1\}^m$ with $d_H(q_c, \operatorname{sign}(Ax)) \leq \beta m$, then any solution $x^\#$ to (7) satisfies $\|x^\# - x\|_2 \leq \rho$.

The results mentioned so far all concern standard Gaussian measurement matrices. For other measurement matrices, signal recovery from the one-bit measurements $q = \operatorname{sign}(Ax)$ can very easily fail, even if the measurement matrix enjoys optimal recovery guarantees in "unquantized" compressed sensing. For instance, it was pointed out in [1] that if $A \in \mathbb{R}^{m\times n}$ is a matrix with entries in $\{-1, 1\}$ (e.g., a Bernoulli matrix), then there are already two-sparse vectors that cannot be accurately recovered. For instance, for any $0 < \lambda < 1$, the vectors

$$x_{+\lambda} = (1 + \lambda^2)^{-1/2}(1, \lambda, 0, \ldots, 0), \qquad x_{-\lambda} = (1 + \lambda^2)^{-1/2}(1, -\lambda, 0, \ldots, 0) \tag{8}$$

produce identical one-bit measurements $\operatorname{sign}(Ax_{+\lambda}) = \operatorname{sign}(Ax_{-\lambda})$, irrespective of the draw of A and the number of measurements. Hence, there is no hope to accurately recover these vectors. Nevertheless, in [1] some non-uniform recovery results from [54] were generalized to subgaussian matrices by imposing additional restrictions. For a fixed $x \in T \subset S^{n-1}$ they showed that $m \gtrsim \rho^{-4} w^2(T)$ suffice to reconstruct x up to error ρ via (7) with high probability provided that either $\|x\|_\infty \le \rho^4$ (since $\|x\|_2 = 1$, this means that the energy of the signal must be sufficiently spread out over its coordinates) or the total variation distance between the subgaussian distribution of the entries of A and the standard Gaussian distribution is at most ρ^{16}.

Even though one-bit compressed sensing generally fails for subgaussian matrices, Foucart [27] identified a different class of matrices for which accurate one-bit compressed sensing is possible. He showed that one can accurately recover signals from one-bit measurements if the measurement matrix satisfies an appropriate RIP-type property of the form (2).

Theorem 3 ([27, Theorem 8]) *If A satisfies $RIP_{1,2}(\Sigma_{2s}, \varepsilon)$, then for every $x \in \mathbb{R}^n$ with $\|x\|_0 \le s$ and $\|x\|_2 = 1$, the hard thresholding reconstruction $x^\#_{HT} = H_s(A^* q)$ satisfies $\|x - x^\#_{HT}\|_2 \le 2\sqrt{5\varepsilon}$.*

Let $\varepsilon \le 1/5$. If A satisfies $RIP_{1,2}(\Sigma^{eff}_{9s}, \varepsilon)$, then for every $x \in \mathbb{R}^n$ with $\|x\|_1 \le \sqrt{s}$ and $\|x\|_2 = 1$, any solution $x^\#_{LP}$ to (6) satisfies $\|x - x^\#_{LP}\|_2 \le 2\sqrt{5\varepsilon}$.

A special case of a result of Schechtman [61] shows that if B is standard Gaussian and $A = \frac{1}{m}\sqrt{\frac{\pi}{2}} B$, then A satisfies $\mathrm{RIP}_{1,2}(T, \varepsilon)$ with probability at least $1 - 2e^{-m\varepsilon^2/2}$ if $m \gtrsim \varepsilon^{-2} w^2(T_n)$, where $T_n = \{x/\|x\|_2 \,:\, x \in T\}$ (see also [55, Lemma 2.1] for a short proof of this special case). In particular, for $T = \Sigma_{2s}$ or $T = \Sigma^{\mathrm{eff}}_{9s}$ this is satisfied if $m \gtrsim \varepsilon^{-2} s \log(en/s)$. Hence, the first statement of Theorem 3 shows that in this case the hard thresholding reconstruction $x^\#_{\mathrm{HT}}$ achieves error ρ if $m \gtrsim \rho^{-4} s \log(en/s)$, which is slightly better than [41, Propositions 1 and 2]. The second statement shows that any solution to the linear program (6) achieves reconstruction error ρ if $m \gtrsim \rho^{-4} s \log(en/s)$, which is a small improvement of the condition originally obtained in [53].

Theorem 3 can be made robust to a small amount of pre-quantization noise: if we observe $q = \mathrm{sign}(Ax + \nu)$, then the first statement holds with error bound $\|x - x^\#_{\mathrm{HT}}\|_2 \lesssim \sqrt{\varepsilon + \|\nu\|_1}$. A similar error bound can be obtained for solutions to an augmented version of the linear program (6), which accounts for the noise. In addition, one can prove a result analogous to Theorem 3 for recovery of low-rank matrices via hard thresholding or a semidefinite program (in the noiseless case, the latter arises by replacing the objective $\|z\|_1$ in (6) by the nuclear norm). We refer to [28] for these extensions and resulting recovery results of low-rank matrices from one-bit standard Gaussian measurements.

In [18], Theorem 3 was used to derive uniform recovery guarantees for randomly subsampled standard Gaussian circulant matrices under a *small sparsity assumption*. For a target reconstruction accuracy $0 < \rho \le 1$, it is assumed that the sparsity s is small enough, i.e.,

$$s \lesssim \rho^2 \sqrt{n/\log(n)}. \tag{9}$$

If $0 < \rho \le (\log^2(s)\log(n))^{-1/4}$ and

$$m \gtrsim \rho^{-4} s \log(en/(s\rho^4))$$

then, with high probability, for any $x \in \mathbb{R}^n$ with $\|x\|_0 \leq s$ and $\|x\|_2 = 1$ the hard thresholding reconstruction $x^{\#}_{\mathrm{HT}}$ satisfies $\|x - x^{\#}_{\mathrm{HT}}\|_2 \leq \rho$. Under slightly stronger conditions a similar uniform reconstruction result can be obtained for effectively sparse vectors on the unit sphere via (6). It is conjectured that a small sparsity assumption is not necessary for these results.

4.2 *Memoryless One-Bit Compressed Sensing With Dithering*

Memoryless one-bit quantization with zero thresholds suffers from two downsides. First, one can only recover signals located on the unit sphere or, viewed differently, only the direction of signals. Second, it is easy to find measurement matrices that perform optimally in "unquantized" compressed sensing for which one-bit compressed sensing fails. These two issues can be resolved by introducing dithering in the quantization process. Let $Q_\tau : \mathbb{R}^m \to \{-1, 1\}^m$ again denote the map $Q_\tau(z) = \mathrm{sign}(z + \tau)$ and consider the measurements $q = Q_\tau(Ax)$. We can interpret this measurement vector geometrically in a similar way as before, except that each measurement now determines a hyperplane $H_{a_i,\tau_i} = \{z \in \mathbb{R}^n \ : \ \langle a_i, z\rangle + \tau_i = 0\}$, which is a parallel shift of the hyperplane H_{a_i}. This immediately explains why dithering can be helpful to recover signals outside of the unit sphere: whereas two signals lying on a straight line cannot be separated by a hyperplane through the origin (and are therefore located in the same cell of the tessellation if $\tau = 0$), they can be separated by shifted hyperplanes. Later we will see that dithering can also greatly extend the class of measurement matrices for which accurate recovery from one-bit measurements can be achieved.

In the setting of Gaussian measurement matrices, recovery results for sparse vectors in the unit ball were first obtained in [4, 43]. In particular, [43] used Gaussian thresholds τ_i and used a slight modification of the linear program (6) for recovery. We will discuss a similar result that was obtained in [4] for the second- order cone program

$$\min_{z\in\mathbb{R}^n} \|z\|_1 \qquad \text{s.t.} \qquad q = \mathrm{sign}(Az + \tau), \ \|z\|_2 \leq R, \tag{10}$$

with $q = \mathrm{sign}(Ax + \tau)$. The rough idea behind the results in [4, 43] is a reduction to the 'standard' one-bit compressed sensing model of Sect. 4.1: we view the dithered measurements $\mathrm{sign}(Ax + \tau)$ as zero-threshold one-bit measurements $\mathrm{sign}([A\ \frac{\tau}{R}]\bar{x})$ of the unit norm vector $\bar{x} = [x, R]/\|[x, R]\|_2 \in S^{n+1}$, where the vector $[x, R] \in \mathbb{R}^{n+1}$ is obtained from x by appending the scalar R as an extra entry. To find an approximant of x, it suffices to find an approximant of $\bar{x}$ of the form $\bar{z}$: by the argument in the proof of [4, Corollary 9] one finds $\|x - z\|_2 \leq 2R\|\bar{x} - \bar{z}\|_2$ for any two vectors $x, z \in RB_{\ell_2^n}$. If A is standard Gaussian, then a small amount of adversarial pre-quantization noise can be handled in a similar fashion by using that A satisfies a *simultaneous* (ℓ_2, ℓ_1)*-quotient* property: with probability at least $1 - e^{-cm}$ any

$\nu \in \mathbb{R}^m$ can be written as $\nu = Au$ for some $u \in \mathbb{R}^n$ with $\|u\|_2 \leq c_1\|\nu\|_2/\sqrt{m}$ and $\|u\|_1 \leq c_1\|\nu\|_2/\sqrt{\log(n/m)}$.

Based on the above reasoning and the binary embedding result (16) stated below, the following was shown.

Theorem 4 ([4, Theorem 2]) *There exist absolute constants c_0, c_1, c_2 such that the following holds. Suppose that $A \in \mathbb{R}^{m\times n}$ is standard Gaussian, $\tau_1, \ldots, \tau_m$ are independent $\mathcal{N}(0, 4R^2)$-distributed. If*

$$m \geq c_0\rho^{-4}s\log(n/s),$$

then the following holds with probability at least $1 - 3e^{-c_1 m\rho^4}$: for any $x \in \mathbb{R}^n$ with $\|x\|_0 \leq s$ and $\|x\|_2 \leq R$ and $q = sign(Ax + \nu + \tau)$ with $\|\nu\|_\infty \leq c_2 R\rho^3$, any solution $x^\#$ to (10) satisfies $\|x - x^\#\|_2 \leq R\rho$.

The linear programming result of [43] and Theorem 4 were extended further to recovery of (effectively) dictionary sparse signals in [3].

Similarly to Theorem 3, uniform recovery via (10) can be ensured via an appropriate RIP$_{1,2}$-property. Suppose that $\nu = 0$ and consider

$$\min_{z\in\mathbb{R}^n} \|z\|_1 \quad \text{s.t.} \quad \text{sign}(C[z, R]) = \text{sign}(C[x, R]), \ \|z\|_2 \leq R, \tag{11}$$

then (10) is obtained by taking $C = [A\ \frac{\tau}{R}]$. It was shown in [18] that if $\varepsilon < 1/5$ and C satisfies RIP$_{1,2}(\Sigma^{\text{eff}}_{36(\sqrt{s}+1)^2}, \varepsilon)$, then for any $x \in \mathbb{R}^n$ satisfying $\|x\|_1 \leq \sqrt{s}\|x\|_2$ and $\|x\|_2 \leq R$, any solution $x^\#$ to (11) satisfies $\|x - x^\#\|_2 \leq 2R\sqrt{\varepsilon}$. To connect this to Theorem 4, note that if τ contains i.i.d. $\mathcal{N}(0, R)$-distributed entries, then $C = [A\ \frac{\tau}{R}]$ is standard Gaussian. By Schechtman's result, $\frac{1}{m}\sqrt{\frac{\pi}{2}}C$ satisfies RIP$_{1,2}(\Sigma^{\text{eff}}_{36(\sqrt{s}+1)^2}, \varepsilon)$ if $m \gtrsim \varepsilon^{-2}s\log(en/s)$ and this immediately implies Theorem 4 (in the case $\nu = 0$). In [18] it was shown that if A is a random partial standard Gaussian circulant matrix, then $\frac{1}{m}\sqrt{\frac{\pi}{2}}C$ with high probability satisfies the same RIP property if $m \gtrsim \varepsilon^{-4}s\log(en/s) + s\log^2 s\log^2 n$ and a certain small sparsity assumption (similar to (9)) is satisfied. Thus, the conclusion of Theorem 4 (for $\nu = 0$) remains valid in this case if $m \gtrsim \rho^{-8}s\log(en/s) + s\log^2 s\log^2 n$.

The program (10) (as well as the linear program in [43]) reconstruct by enforcing quantization consistency. For this reason, this program can easily fail in the case of post-quantization noise, as has been discussed in Sect. 4.1. In addition, since the approaches in [4, 18, 43] essentially reduce to the standard one-bit compressed sensing model, the type of measurement matrices for which results can be obtained is relatively limited: so far only reconstruction results are known for standard Gaussian and, under additional restrictions, randomly subsampled standard Gaussian circulant matrices and subgaussian matrices. These limitations were overcome in [20, 21] by using uniform dithering, as we will now discuss.

In [20], recovery results were obtained for matrices with i.i.d. subgaussian or heavy-tailed rows, which are stable and robust with respect to both pre- and post-quantization noise. Suppose that we observe a vector $q_c \in \{-1, 1\}^m$ satisfying

$$d_H(q_c, \operatorname{sign}(Ax + \nu + \tau)) \leq \beta m,$$

i.e., at most a fraction β of the bits are arbitrarily corrupted during quantization. Consider

$$\min_{z \in \mathbb{R}^n} d_H(q_c, \operatorname{sign}(Az + \tau)) \qquad \text{s.t.} \qquad z \in T. \tag{12}$$

This (non-convex) program selects an $x^\#$ whose noiseless one-bit measurements minimize the Hamming distance to the corrupted vector of quantized noisy measurements. The following recovery theorem applies to subgaussian random matrices. A more general version of this result can be proved for *heavy-tailed* measurement vectors, see [20].

Theorem 5 ([20, Theorem 1.5]) *There exist constants $c_0, \ldots, c_4 > 0$ depending only on L such that the following holds. Suppose that $A \in \mathbb{R}^{m \times n}$ has i.i.d. symmetric, isotropic, L-subgaussian rows, ν has i.i.d. L-subgaussian entries with variance σ^2, and τ has i.i.d. entries which are uniformly distributed on $[-\lambda, \lambda]$. Let $T \subset RB_2^n$, set $\lambda \geq c_0(R + \sigma) + \rho$, put $r = c_1\rho/\sqrt{\log(e\lambda/\rho)}$, and let $T_r = (T - T) \cap rB_2^n$. Assume that*

$$m \geq c_2 \lambda \left(\frac{w^2(T_r)}{\rho^3} + \frac{\log \mathcal{N}(T, r)}{\rho} \right), \tag{13}$$

and that $|\mathbb{E}\nu_1| \leq c_3\rho$, $\sigma \leq c_3\rho/\sqrt{\log(e\lambda/\rho)}$ and $\beta \leq c_3\rho/\lambda$. Then, with probability at least $1 - 10\exp(-c_4 m\rho/\lambda)$, for every $x \in T$, any solution $x^\#$ of (12) satisfies $\|x^\# - x\|_2 \leq \rho$.

If $T \subset B_2^n$ and $\sigma \leq 1$ then λ is a constant that depends only on L. In this case (see [20] for details) (13) holds if

$$m = c(L) \frac{\log(e/\rho)}{\rho^3} w^2(T).$$

In the special case $T = \Sigma_s \cap B_2^n$, a much better estimate is possible:

$$m = c'(L)\rho^{-1} s \log\left(\frac{en}{s\rho}\right).$$

The latter is optimal in terms of s and n and optimal up to the log-factor in terms of ρ.

The result in Theorem 5 is still rather sensitive to pre-quantization noise: the mean and variance of the noise should be of the order of ρ. In addition to this sensitivity to pre-quantization noise, the program (12) is computationally hard to solve. To resolve these two issues a different program, which is essentially obtained by convexifying the objective of (12), was introduced in [20]: for $\lambda > 0$ consider

$$\max_{z \in \mathbb{R}^n} \frac{1}{m}\langle q_c, Az \rangle - \frac{1}{2\lambda}\|z\|_2^2 \qquad \text{s.t.} \qquad z \in U, \tag{14}$$

where either $U = T$ or $U = \operatorname{conv}(T)$. In the first case, we can view (14) as a regularized version of (7). As is pointed out in [21], (14) is equivalent to

$$\min \left\| z - \frac{\lambda}{m} A^* q_c \right\|_2 \qquad \text{s.t.} \qquad z \in U, \tag{15}$$

i.e., it computes an ℓ_2-projection $P_U(\frac{\lambda}{m} A^* q_c)$ of $\frac{\lambda}{m} A^* q_c$ onto U. If $U = \operatorname{conv}(T)$, then (14) is convex, has a unique solution and can be expected to be stable. On the other hand, if T is "simple", then it may be advantageous to take $U = T$. For instance, if $U = T = \Sigma_s \cap B_2^n$, then (14) has a closed-form solution

$$x^\# = \min\left\{\frac{\lambda}{m}, \frac{1}{\|H_s(A^* q_c)\|_2}\right\} H_s(A^* q_c),$$

where H_s is the hard thresholding operator. The following result is stated for $U = \operatorname{conv}(T)$ in [20, Theorem 1.7], the case $U = T$ is immediate from the proof.

Theorem 6 ([20, Theorem 1.7]) *There exist constants $c_0, \ldots, c_4$ that depend only on L for which the following holds. Suppose that either $U = T$ and $T - T$ is star-shaped or $U = \operatorname{conv}(T)$. Suppose that A has i.i.d. symmetric, isotropic, L-subgaussian rows, ν has i.i.d. mean-zero, L-subgaussian entries with variance σ, and τ has i.i.d. entries which are uniformly distributed on $[-\lambda, \lambda]$. Let $T \subset RB_2^n$, fix $\rho > 0$, set $U_\rho = (U - U) \cap \rho B_2^n$,*

$$\lambda \geq c_0(\sigma + R)\sqrt{\log(c_0(\sigma + R)/\rho)}$$

and let $r = c_1\rho/\log(e\lambda/\rho)$. If m and β satisfy

$$m \geq c_2\left(\left(\frac{\lambda w(U_\rho)}{\rho^2}\right)^2 + \lambda^2 \frac{\log \mathcal{N}(T, r)}{\rho^2}\right), \qquad \beta\sqrt{\log(e/\beta)} = c_3\frac{\rho}{\lambda},$$

then, with probability at least $1 - 8\exp(-c_4 m\rho^2/\lambda^2)$, for any $x \in T$ any solution $x^\#$ of (14) satisfies $\|x^\# - x\|_2 \leq \rho$.

If T is the set of sparse or compressible vectors in RB_2^n, then Theorem 6 can be extended to randomly subsampled subgaussian circulant matrices (with rows selected according to the selector model). The only difference is that some additional logarithmic factors appear in the result. We refer to [21, Theorem 1.1] for details.

If $T = \Sigma_s \cap B_2^n$ and $\sigma \geq 1$, then we can take $\lambda = c(L)\sigma\sqrt{\log(c(L)\sigma/\rho)}$ and

$$m = c'(L)\frac{\sigma^2}{\rho^2} s \log\left(\frac{\sigma}{\rho}\right)\left(\log\left(\frac{en}{s\rho}\right) + \log\log\left(\frac{e\sigma}{\rho}\right)\right),$$

which is optimal up to logarithmic factors by Theorem 1.

4.3 Relation to Binary Embeddings

The robust recovery result in Theorem 2 relies on a beautiful geometric result due to Plan and Vershynin [55]. They showed that if $T \subset S^{n-1}$, $m \gtrsim \rho^{-6} w^2(T)$, and $A \in \mathbb{R}^{m\times n}$ is a standard Gaussian matrix then, with probability at least $1 - 2e^{-cm\rho^2}$, for all $x, y \in T$,

$$d_{S^{n-1}}(x, y) - \rho \le \frac{1}{m} d_H(\mathrm{sign}(Ax), \mathrm{sign}(Ay)) \le d_{S^{n-1}}(x, y) + \rho. \tag{16}$$

In other words, if x and y are "separated enough", then the fraction of the random Gaussian hyperplanes $H_{a_i} = \{z \in \mathbb{R}^n : \langle a_i, z\rangle = 0\}$ that separate x and y approximates their geodesic distance in a very sharp way. It was later shown in [52] that (16) remains true if $m \gtrsim \rho^{-4} w^2(T)$. Moreover, for certain "simple" sets (e.g., if T is the set of unit norm sparse vectors) it is known that $m \gtrsim \rho^{-2} w^2(T)$ suffices for (16) (see [42, 52, 55] for examples).

In a similar way, the reconstruction results in Theorems 5 and 6 are connected to "isomorphic" versions of (16). To give a concrete example from [20], suppose that A has i.i.d. symmetric, isotropic, L-subgaussian rows and that the entries of τ are i.i.d. uniformly distributed on $[-\lambda, \lambda]$. If $T \subset RB_2^n$, $\lambda = c_0 R$ and

$$m \ge c_1 \frac{R \log(eR/\rho)}{\rho^3} w^2(T),$$

then with probability at least $1 - 8\exp(-c_2 m\rho/R)$, for any $x, y \in \mathrm{conv}(T)$ such that $\|x - y\|_2 \ge \rho$, one has

$$c_3 \frac{\|x - y\|_2}{R} \le \frac{1}{m} d_H(\mathrm{sign}(Ax + \tau), \mathrm{sign}(Ay + \tau)) \le c_4 \sqrt{\log(eR/\rho)} \cdot \frac{\|x - y\|_2}{R}, \tag{17}$$

where $c_0, \ldots, c_4$ depend only on L. Hence, if x and y are separated enough, then the fraction of the hyperplanes $H_{a_i,\tau_i} = \{v \in \mathbb{R}^n : \langle a_i, v\rangle + \tau_i = 0\}$ that separate x and y accurately approximates their Euclidean distance.

4.4 Exponential Error Decay Via Adaptive Thresholds

Let us now briefly discuss how one can achieve optimal, exponential error decay in terms of the number of measurements by using adaptive thresholds, following the idea put forward in [4]. Interestingly, this scheme completely integrates the analog measurement, quantization, and reconstruction procedures. Our presentation follows [22].

To sketch the high-level idea, recall that in memoryless one-bit compressed sensing, by taking measurements we geometrically produce hyperplanes through the origin (if $\tau = 0$) or shifted versions thereof ($\tau \neq 0$). In both cases, the origin is our "reference point" for producing hyperplanes. Intuitively, this is a good strategy to locate x if x happens to lie close to the origin, but relatively ineffective if x is far away. This is reflected by the appearance of the radius R of the signal set in the reconstruction results discussed in Sect. 4.2. To improve the error decay, we can proceed as follows: we first take a small batch of memoryless quantized measurements and run a reconstruction algorithm to obtain a rough estimate $\hat{x}$ of the location of x. In the next round, we use $\hat{x}$ as a new reference point to produce hyperplanes. Continuing in this fashion, we "move in" on the target signal x and are able to produce more informative measurements in each round.

Formally, fix a closed signal set $K \subset \mathbb{R}^n$ with $0 \in K$ and let P_K be the ℓ_2-projection onto this set. We set $I_i = \{(i-1)m/B + 1, \dots, im/B\}$ and divide the measurement matrix A into the submatrices $A_{(i)} = R_{I_i} A$, $1 \le i \le B$, each containing m/B consecutive rows of A. We let $\nu_{(i)} = R_{I_i}\nu$, $\tau_{(i)} = R_{I_i}\tau$ and $R_i = 2^{-i}R$. Suppose that we have an algorithm $\mathscr{A}_i$ which, with probability at least $1-\eta$, satisfies the following for any $w \in K - K$ with $\|w\|_2 \le R_{i-1}$: based on the input $(A_{(i)}, \tau_{(i)}, (q_c)_{(i)}, R_{i-1})$, with $(q_c)_{(i)} \in \{-1,1\}^{m/B}$ satisfying for a certain $\bar{\tau}_{(i)} = \bar{\tau}_{(i)}(A_{(i)}, \tau_{(i)}, R_{i-1}) \in \mathbb{R}^{m/B}$

$$d_H((q_c)_{(i)}, \operatorname{sign}(A_{(i)}w + \nu_{(i)} + \bar{\tau}_{(i)})) \le \beta m/B, \tag{18}$$

$\mathscr{A}_i$ produces a $w^\# \in \mathbb{R}^n$ so that $\|w - w^\#\|_2 \le R_{i-1}/4$. We can then produce partial reconstructions $(\bar{x}_{(i)})_{i=1}^B$ of x iteratively as follows. Suppose that we produced an $\bar{x}_{(i-1)} \in K$ satisfying $\|x - \bar{x}_{(i-1)}\|_2 \le R_{i-1}$. We acquire corrupted measurements $(q_c)_{(i)}$ satisfying (18) for $w = x - \bar{x}_{(i-1)}$. Since

$$\operatorname{sign}(A_{(i)}w + \nu_{(i)} + \bar{\tau}_{(i)}) = \operatorname{sign}(A_{(i)}x + \nu_{(i)} + \mu_{(i)} + \bar{\tau}_{(i)}),$$

with $\mu_{(i)} = -A_{(i)}\bar{x}_{(i-1)}$, the desired $(q_c)_{(i)}$ can be acquired by measuring x with $A_{(i)}$ and using $Q_{(\mu_{(i)}+\bar{\tau}_{(i)})}$ as a quantizer.

We now input $(A_{(i)}, \tau_{(i)}, (q_c)_{(i)}, R_{i-1})$ into the algorithm $\mathscr{A}_i$ and let $x^\#_{(i)}$ be its output. Define $\bar{x}_{(i)} = P_K(\bar{x}_{(i-1)} + x^\#_{(i)})$. Clearly, since $x \in K$,

$$\begin{aligned}\|x - \bar{x}_{(i)}\|_2 &\le \|x - \bar{x}_{(i-1)} - x^\#_{(i)}\|_2 + \|\bar{x}_{(i-1)} + x^\#_{(i)} - P_K(\bar{x}_{(i-1)} + x^\#_{(i)})\|_2 \\ &\le 2\|x - \bar{x}_{(i-1)} - x^\#_{(i)}\|_2 \le 2\frac{R_{i-1}}{4} = R_i.\end{aligned}$$

Hence, if $\|x\|_2 \le R$ and we set $\bar{x}_{(0)} = 0$, then by induction we find $\|x - \bar{x}_{(i)}\|_2 \le R2^{-i}$ for all $1 \le i \le B$. In summary, if we set $B = \log_2(R/\rho)$ then, with probability at least $1 - B\eta$, $\|x - \bar{x}_{(B)}\|_2 \le \rho$ for any $x \in K$.

In the original paper [4], recovery results with exponential error decay were obtained via the above scheme for s-sparse vectors and standard Gaussian measurement matrices using either hard thresholding operations or Gaussian dithering and the second-order cone program (10) to produce partial reconstructions. In [28], these results were extended to recovery of low-rank matrices, using either hard thresholding or a semidefinite program. In [21, 22], an exponential decay scheme was derived for sparse vectors and randomly subsampled subgaussian circulant matrices using uniform dithering and hard thresholding for partial reconstruction.

As a variation of the result in [21, 22], we will derive a general result valid for any signal set K which is a closed cone, any $A \in \mathbb{R}^{m\times n}$ with i.i.d. symmetric, isotropic, L-subgaussian rows, and uniform dithering. We only need to specify the "base algorithms" $\mathscr{A}_i$. We consider a $w \in (K-K) \cap R_{i-1}B_2^n$ and acquire measurements $(q_c)_{(i)}$ satisfying

$$d_H((q_c)_{(i)}, \operatorname{sign}(A_{(i)}w + \nu_{(i)} + \bar{\tau}_{(i)})) \leq \beta m/B$$

with $A_{(i)} = R_{I_i}A$, $\nu_{(i)} = R_{I_i}\nu$, $\tau_{(i)} = R_{I_i}\tau$, and $\bar{\tau}_{(i)} = R_{i-1}\tau_{(i)}$, where τ has i.i.d. entries which are uniformly distributed on $[-\lambda, \lambda]$. Clearly,

$$\begin{aligned} &d_H((q_c)_{(i)}, \operatorname{sign}(A_{(i)}w + \nu_{(i)} + \bar{\tau}_{(i)})) \\ &\quad = d_H((q_c)_{(i)}, \operatorname{sign}(A_{(i)}(w/R_{i-1}) + \nu_{(i)}/R_{i-1} + \tau_{(i)})). \end{aligned}$$

Define $\tilde{w} = P_{(K-K)\cap B_2^n}(\frac{\lambda}{m}A_{(i)}^*(q_c)_{(i)})$. Since K is a cone, $w/R_{i-1} \in (K-K)\cap B_2^n$. Hence, Theorem 6 (applied with $T = (K-K)\cap B_2^n$, $\rho = 1/4$, and $R = 1$) shows that if we assume that ν contains i.i.d. mean-zero, L-subgaussian entries with variance $\sigma^2 \leq \rho^2 \leq R_{i-1}^2$ and set

$$m/B \geq c_1 w^2((K-K)\cap B_2^n), \qquad \lambda = c_2, \qquad \beta\sqrt{\log(e/\beta)} = c_3,$$

then, with probability at least $1 - 8\exp(-c_4 m/B)$, for all $w \in (K-K)\cap R_{i-1}B_2^n$ the vector $\tilde{w}$ satisfies $\|\frac{w}{R_{i-1}} - \tilde{w}\|_2 \leq 1/4$. Hence, the vector $w^\# = R_{i-1}\tilde{w}$ has the desired properties.

Our considerations lead to the following algorithm and result.

Algorithm 1: exponentially decaying scheme

```
Input: $A \in \mathbb{R}^{m\times n}$, $B \in \mathbb{N}$, $\tau \in \mathbb{R}^m$, $R > 0$
Initialization: $\bar{x}_{(0)} = 0$.
for i=1,...,B do
    $A_{(i)} = R_{I_i}A$
    $\mu_{(i)} = -A_{(i)}\bar{x}_{(i-1)}$
    $\nu_{(i)} = R_{I_i}\nu$
    $\tau_{(i)} = \mu_{(i)} + R2^{-(i-1)}R_{I_i}\tau$
    Produce corrupted quantized measurements $(q_c)_{(i)} \in \{-1,1\}^{m/B}$ with
        $d_H((q_c)_{(i)}, \operatorname{sign}(A_{(i)}x + \nu_{(i)} + \tau_{(i)})) \leq \beta m/B$
    $x_{(i)}^\# = R2^{-(i-1)}P_{(K-K)\cap B_2^n}\left(\frac{\lambda}{m}A_{(i)}^*(q_c)_{(i)}\right)$
    $\bar{x}_{(i)} = P_K(\bar{x}_{(i-1)} + x_{(i)}^\#)$
end
Output: $x^\# = \bar{x}_{(B)}$
```

Theorem 7 *There exist constants c_1, c_2, c_3, c_4 depending only on L such that the following holds. Let $K \subset \mathbb{R}^n$ be a closed cone, fix $0 < \rho \leq 1$ and $R > 0$, set $B = \log_2(R/\rho)$, $\lambda = c_1$, $m \geq c_2 B w^2((K-K) \cap B_2^n)$, $\beta = c_3$. Suppose that A has i.i.d. symmetric, isotropic, L-subgaussian rows, ν has i.i.d. mean-zero, L-subgaussian entries with variance $\sigma \leq \rho$, τ has i.i.d. entries which are uniformly distributed on $[-\lambda, \lambda]$, and A, ν, τ are independent. Then with probability at least $1 - Be^{-c_4 m/B}$ the following holds: for any $x \in K$ with $\|x\|_2 \leq R$, the output $x^{\#}$ of Algorithm 1 satisfies $\|x - x^{\#}\|_2 \leq \rho$.*

The decay of the reconstruction error in Theorem 7 is clearly superior to the error decay in Theorem 6. The total number of measurements generated in Algorithm 1 is

$$m \sim \log(R/\rho) w^2((K-K) \cap B_2^n),$$

so the reconstruction error decays exponentially in terms of the number of measurements, which is optimal (see the discussion in Sect. 2). In addition, the total number of adversarial bit corruptions is βm, a constant fraction of m.

The price to pay for this superior scheme is more complicated hardware and higher energy consumption in operation. The quantizer needs to be equipped with memory and the capability to compute and set new thresholds in each round.

5 Memoryless Multi-bit Compressed Sensing

Let us now consider memoryless multi-bit quantization schemes. A *memoryless scalar quantizer* is defined by fixing a quantization alphabet $\mathcal{Q} \subset \mathbb{R}$ and setting, for a given $z \in \mathbb{R}^m$ and $i \in [m]$,

$$Q_{\mathrm{MSC}}(z)_i = \min\{\mathrm{argmin}_{t \in \mathcal{Q}} |z_i - t|\}.$$

For example, by taking the alphabet $\mathcal{Q} = \{-1, 1\}$ we find the one-bit quantizer with zero thresholds studied in Sect. 4.1. Before discussing specific recovery algorithms, let us first point out that the best reconstruction error decay in terms of the number of measurements that *any* reconstruction algorithm can achieve when receiving memoryless scalar quantized measurements as input is, in general, *linear*. Specifically, it was shown in [6, 30] that if $A \in \mathbb{R}^{m \times n}$ and $E \subset \mathbb{R}^n$ is a k-dimensional subspace, then $\sup_{x \in E} \|x - \mathscr{A}(Q_{\mathrm{MSC}}(Ax))\|_2 \geq c\frac{k}{m}$ for any reconstruction map $\mathscr{A} : \mathbb{R}^m \to \mathbb{R}^n$.

The most studied memoryless multi-bit compressed sensing model involves the memoryless scalar quantizer with alphabet $\mathcal{Q} = \delta\mathbb{Z}$, i.e., the quantizer $Q_\delta : \mathbb{R}^m \to (\delta\mathbb{Z})^m$ defined by

$$Q_\delta(z) = \left(\delta \lfloor z_i/\delta \rfloor\right)_{i=1}^m.$$

For brevity, we will call this map the *uniform scalar quantizer*. Geometrically, Q_δ divides $\mathbb{R}^m$ into half-open cubes with side lengths equal to δ and maps any vector

$z \in \mathbb{R}^m$ to the corner of the cube in which it is located. From a practical point of view, this quantizer is somewhat idealized: in a realistic implementation the range of the quantizer is limited and measurements $\langle a_i, x\rangle$ which exceed the quantizer's range incur a potentially unbounded quantization error. One calls such measurements *saturated*. The work [46] analyzes some strategies to deal with saturated measurements. We will restrict ourselves to the idealized uniform scalar quantizer.

Let us first consider the "agnostic" approach to reconstruct x from uniformly scalar quantized measurements $q = Q_\delta(Ax)$, i.e., we simply treat the error due to quantization as additive noise. Note that the ℓ_∞-distance of Ax to the center of its quantization cell, i.e., $q + (\delta/2)\mathbb{1}$ where $\mathbb{1} \in \mathbb{R}^m$ is the vector which has all entries equal to 1, is at most $\delta/2$. Hence, we can reconstruct the signal x via the linear program

$$\min_{z\in\mathbb{R}^n} \|z\|_1 \qquad \text{s.t.} \qquad \|Az - (q + (\delta/2)\mathbb{1})\|_\infty \leq \delta/2. \tag{19}$$

Note that this method is very close to minimizing the ℓ_1-norm under a quantization consistency constraint, i.e., to solving

$$\min \|z\|_1 \qquad \text{s.t.} \qquad y = Q_\delta(Az). \tag{20}$$

Indeed, whereas z is feasible for (20) if and only if Az lies in the same quantization cell as Ax, z is feasible for (19) precisely when it lies in the closure of that cell.

From the standard theory of compressed sensing, it is easy to extract (see [18, Theorem A.1]) that if $A \in \mathbb{R}^{m\times n}$ is such that $\frac{1}{\sqrt{m}}A$ satisfies $\mathrm{RIP}_{2,2}(\Sigma_s, c)$ with constant $c < 4/\sqrt{41}$, then for any $x \in \mathbb{R}^n$ and $y = Q_\delta(Ax)$ any solution $x^\#$ to (19) satisfies

$$\|x - x^\#\|_2 \lesssim \delta + s^{-1/2} \inf_{z\in\Sigma_s} \|x - z\|_1. \tag{21}$$

In particular, this applies to partial subgaussian circulant matrices (with deterministically selected rows) if $m \gtrsim s\log^2 s \log^2 n$ [44] and randomly subsampled discrete bounded orthonormal systems if $m \gtrsim s\log^2 s \log n$ [35]. A different argument, which relies on Mendelson's small ball method [48] instead of an RIP-based analysis, shows that even for a variety of heavy-tailed random matrices the reconstruction guarantee (21) holds in the optimal regime $m \gtrsim s\log(en/s)$ (see [19, Section V] for several results).

Although these results exhibit the same dependence of m on s and n as in "unquantized" compressed sensing, they have a clear downside: by treating the quantization error as noise, the reconstruction error does not decay beyond the resolution δ of the quantizer, which corresponds to the noise floor. Intuitively, one could hope to be able to decrease the reconstruction error even beyond the resolution δ by taking more measurements. In a series of works by L. Jacques and co-authors [37, 39, 40, 63], this is shown to be possible if one introduces appropriate dithering at the quantizer. Let us denote by $Q_{\delta,\tau} = Q_\delta(\cdot + \tau)$ the uniform scalar quantizer with dithering vector $\tau \in \mathbb{R}^m$. It was first observed in [37] (see also [63, Appendix A]) that if the entries

τ_i of τ are i.i.d. uniformly distributed on $[0, \delta]$, then for any $y \in \mathbb{R}^m$, $\mathbb{E}Q_{\delta,\tau}(y) = y$. Hence, at least in expectation, dithering that matches the resolution can "cancel out" the error caused by the uniform scalar quantizer. This fact can be exploited to prove recovery results for general signals sets and a large class of measurement matrices. We start by describing a result from [63]. Let $T \subset \mathbb{R}^n$ be a closed set of signals. For $x \in T$ consider its quantized measurements $q = Q_{\delta,\tau}(Ax)$ and define

$$x^{\#}_{\mathrm{PBP}} = P_T\Big(\frac{1}{m}A^*q\Big).$$

Since A^*q is usually called the back projection of q, this reconstruction is coined the *projected back projection* in [63]. If $T = \Sigma_s$, then the projected back projection is up to scaling the same as the hard thresholding map in Theorem 3. To give the flavor of the recovery results in [63], we state a recovery result if T is a union of subspaces. Further results are obtained for low-rank matrices and star-shaped convex sets.

Theorem 8 ([63]) *Let $T = \cup_{i=1}^N T_i \subset \mathbb{R}^n$ be a union of subspaces. Suppose that the entries of τ are i.i.d. uniformly distributed on $[0, \delta]$. Let $A \in \mathbb{R}^{m\times n}$ be a random matrix that, for any fixed $0 < \varepsilon < 1$, satisfies*

$$\Big|\frac{1}{m}\|Az\|_2^2 - \|z\|_2^2\Big| \le \varepsilon, \qquad \textit{for all } z \in T \cap B_2^n$$

with probability at least $1 - \eta$ if

$$m \gtrsim \varepsilon^{-2} w^2(T \cap B_2^n)\,\mathrm{polylog}(m, n, 1/\eta).$$

Let $T^{(4)} = \sum_{i=1}^4 T$. If $m \gtrsim \rho^{-2}(1+\delta)^2 w^2(T^{(4)} \cap B_2^n)\,\mathrm{polylog}(m, n, \delta, 1/\rho, 1/\eta)$, then with probability at least $1-\eta$, for any $x \in T \cap B_2^n$ the projected back projection $x^{\#}_{PBP}$ satisfies $\|x - x^{\#}\|_2 \le \rho$.

In the special case $T = \Sigma_s$, the assumption of Theorem 8 is e.g. satisfied if A is subgaussian, a partial subgaussian circulant matrix or a randomly subsampled discrete bounded orthonormal system. Hence, for these matrices, one can uniformly recover all s-sparse vectors from their projected back projections if $m \gtrsim \rho^{-2}(1+\delta)^2 s \log(en/s)\,\mathrm{polylog}(m, n, \delta, 1/\rho, 1/\eta)$.

The reconstruction error in Theorem 8 does not decrease to zero as the bin width δ goes to zero, as e.g. in (21). In fact, this cannot be expected as $x^{\#}_{\mathrm{PBP}}$ will, loosely speaking, start behaving as $H_s(\frac{1}{m}A^*Ax)$ as $\delta \to 0$, i.e., as the first step of the iterative hard thresholding algorithm in "unquantized" compressed sensing. Therefore, it is of interest to derive a "best of both worlds" result that exhibits both a decaying reconstruction error in terms of the number of measurements and, at the same time, a reconstruction error decaying to zero if $\delta \to 0$ once m exceeds the threshold of $Cs\log(en/s)$ measurements, which are needed for uniform recovery from unquantized measurements. One can get very close to such a result by using a relation between uniform scalar quantization and so-called *quantized Johnson-Lindenstrauss* *embeddings*. This relation is analogous to the connection between one-bit compressed sensing and binary embeddings sketched in Sect. 4.3. For concreteness, we consider the following embedding result.

Theorem 9 ([40, Proposition 1]) *If $m \gtrsim \varepsilon^{-2}\log N(T,\delta\varepsilon^2)$ and $\frac{1}{m}A \in \mathbb{R}^{m\times n}$ satisfies RIP$_{1,2}(T-T,\theta)$, then for certain absolute constants $c, C > 0$, with probability at least $1 - Ce^{-cm\varepsilon^2}$ the map $f(x) = Q_{\delta,\tau}(Ax)$ satisfies*

$$(1-\theta)\|x-y\|_2 - c\delta\varepsilon \leq \frac{1}{m}\|f(x)-f(y)\|_1 \leq (1+\theta)\|x-y\|_2 + c\delta\varepsilon \qquad (22)$$

for all $x, y \in T$.

By the lower bound in (22), for any given signal $x \in T$, any $x^\# \in T$ that is quantization consistent with x satisfies $\|x - x^\#\|_2 \leq c\delta\varepsilon/(1-\theta)$. Thus, under the conditions of Theorem 9 we can recover x via a program that finds a quantization consistent vector in T. In particular, if $T = \Sigma_s \cap B_2^n$ then we can use the non-convex program

$$\min \|z\|_0 \qquad \text{s.t.} \qquad q = Q_{\delta,\tau}(Az), \qquad \|z\|_2 \leq 1. \qquad (23)$$

If B is standard Gaussian and $A = \sqrt{\frac{\pi}{2}}B$, then $\frac{1}{m}A$ satisfies RIP$_{1,2}(\Sigma_{2s},\theta)$ with probability at least $1 - 2e^{-cm\theta^2}$ if $m \gtrsim \theta^{-2}s\log(en/s)$. Combining this fact with Theorem 9 and the estimate $\log N(\Sigma_s \cap B_2^n, \delta\varepsilon^2) \lesssim s\log(en/(s\delta\varepsilon^2))$, we find that if $m \gtrsim \varepsilon^{-2}s\log(en/(s\delta\varepsilon^2))$, then with probability at least $1 - Ce^{-cm\varepsilon^2}$, for any $x \in \Sigma_s \cap B_2^n$, any solution $x^\#$ to (23) satisfies $\|x - x^\#\|_2 \leq \delta\varepsilon$.

This result can still be improved, since to derive a recovery result it suffices to prove a much weaker property than (22). In [38, 39] a direct analysis was made of the required property

$$Q_{\delta,\tau}(Az) = Q_{\delta,\tau}(Ax) \Rightarrow \|x - z\|_2 \leq \theta, \qquad \text{for all } x, z \in T. \qquad (24)$$

If (24) holds for $T = \Sigma_s \cap B_2^n$ and $\theta = \delta\varepsilon$, then for any $x \in \Sigma_s \cap B_2^n$ any solution $x^\#$ to (23) satisfies $\|x^\# - x\|_2 \leq \delta\varepsilon$. It was shown in [38, Theorem 2] that a standard Gaussian matrix $A \in \mathbb{R}^{m\times n}$ satisfies this property with high probability if $m \gtrsim \varepsilon^{-1}s\log(en/(\sqrt{s}\delta\varepsilon))$. Since for a fixed δ the reconstruction error cannot decay faster than linear in m, the dependence of m on ε is near-optimal in this result.

We refer to [39, 40] for further results on quantized Johnson-Lindenstrauss embeddings, in particular versions involving RIP$_{2,2}$-matrices and subgaussian matrices, and to [38, 39] for further results concerning the property (24). The latter results are used in [51] to derive reconstruction guarantees for generalizations of (23) in which $\|z\|_0$ is replaced by an atomic norm.

In [18], Theorem 9 was used to prove a uniform recovery result for effectively s-sparse vectors in the unit ball from randomly subsampled Gaussian circulant measurements (with rows selected according to the selector model) via a convex program that enforces quantization consistency. Loosely speaking, [18, Theorem 6.2] shows that with high probability one can achieve a reconstruction error $\varepsilon\delta^{2/3}$ using roughly

$m \sim \varepsilon^{-6} s \log(en/s)$ measurements, provided that a small sparsity condition is satisfied. Interestingly, this result uses a combination of Gaussian and uniform dithering in the quantizer.

6 Noise-Shaping Methods

Finally, we discuss quantized compressed sensing with a family of adaptive quantization methods called *noise-shaping methods*. The most prominent example in this family are $\Sigma\Delta$-quantization methods, which are very popular in practice. Noise-shaping quantizers were first studied mathematically in the context of analog-to-digital conversion of bandlimited functions (see e.g., [14, 33]) and afterwards have been successfully extended to the frameworks of finite frames and compressed sensing (see e.g., the survey [13] and the references therein). In the setting of compressed sensing, the first reconstruction results for exactly sparse signals were obtained via a two-stage approach [25, 34, 45]. First, one estimates only the support of the original sparse signal via a traditional compressed sensing method for noisy measurements. Once the support is known, one can use reconstruction methods developed in the framework of finite frames to fully reconstruct the signal, e.g., by using an appropriate Sobolev dual frame. For the sake of brevity, we will not discuss this approach and refer to the survey [13] for details. We will only discuss a recent one-stage recovery approach via a convex program, which was developed in [10, 13, 26, 36, 60]. In contrast to the two-stage approach sketched above, this method is proven to be stable with respect to approximate sparsity, robust with respect to (a small amount of) pre-quantization noise and has been successfully applied to structured random measurement matrices [26, 36].

A *noise-shaping quantizer* $Q : \mathbb{R}^m \to \mathcal{Q}^m$ associated with a *noise transfer operator* H, is defined so that for each $y \in \mathbb{R}^m$ the quantization $q = Q(y)$ satisfies the *noise-shaping relation*

$$y - q = Hu \tag{25}$$

where $u = u(y, Q) \in \mathbb{R}^m$ is an auxiliary vector called the *internal state vector*. The matrix $H \in \mathbb{R}^{m \times m}$ is chosen to be a lower triangular Toeplitz matrix with unit diagonal, so that the quantization scheme can be implemented via a recursion. The noise-shaping quantizer is called *stable* if, for all $y \in \mathbb{R}^m$ with $\|y\|_\infty \leq \mu$, $\|u\|_\infty \leq C_{Q,\mu}$, where $C_{Q,\mu}$ is a constant independent of m called the *stability constant*. The most important examples of noise-shaping quantizers are *$\Sigma\Delta$-quantizers*, which compute a solution to the noise-shaping relation (25) for $H = D^r$, where $D \in \mathbb{R}^m$ is the first-order difference matrix defined by

$$D_{ij} = \begin{cases} 1 & \text{if } i = j \\ -1 & \text{if } i = j + 1 \\ 0 & \text{else.} \end{cases}$$

We call r the *order* of the scheme. The construction of a stable r-th order $\Sigma\Delta$-scheme is non-trivial. It was shown in [16] that for any $L \in \mathbb{N}$ and $\delta > 0$ there exists a stable r-th order $\Sigma\Delta$-scheme with a fixed alphabet $\mathcal{Q}_{\delta,L} = \{\pm(2\ell - 1)\delta \ : \ 1 \leq \ell \leq L\}$ and constant

$$C_{\mathcal{Q},\mu} \leq C\delta \left(\frac{er}{\pi} \left\lceil \frac{\pi^2}{(\cosh^{-1}(2L - \frac{\mu}{\delta}))^2} \right\rceil \right)^r .$$

In particular, taking $L = 1$, $\delta = 1$, we find an r-th order scheme with the one-bit alphabet $\mathcal{Q} = \{-1, 1\}$ which is stable in the sense that $\|u\|_\infty \leq Cc_\mu^r r^r$ whenever $\|y\|_\infty \leq \mu < 1$.

Let us now turn to the compressed sensing scenario, where $y = Ax$ and the noise-shaping relation is

$$Ax - q = Hu.$$

To see how we could recover x, multiply both sides by a designed preconditioning matrix $V \in \mathbb{R}^{p\times m}$ to obtain

$$VAx - Vq = VHu.$$

Since we observe Vq, we can interpret this equation as a linear measurement equation $Vq = VAx + e$, where VA is the measurement matrix and $e = -VHu$ is the noise on the measurements. To recover x, we can then use methods for recovery from "unquantized" noisy measurements. For instance, we can use basis pursuit denoising

$$\min_{z\in\mathbb{R}^n} \|z\|_1 \qquad \text{s.t.} \qquad \|VAz - Vq\|_2 \leq \eta. \tag{26}$$

By a standard result in compressed sensing, one can recover any s-sparse x via (26) if VA satisfies $\mathrm{RIP}_{2,2}(\Sigma_s, c)$ for c a small enough absolute constant and $\|e\|_2 \leq \eta$ (see e.g., [29, Chapter 9]). To satisfy the latter condition, if we assume that the quantization scheme is stable and $\|Ax\|_\infty \leq \mu$, it suffices to ensure that $\|VH\|_{\ell_\infty\to\ell_2}$ is small.

In the presence of pre-quantization noise, the noise-shaping relation changes to

$$V(Ax + \nu) - Vq = VHu.$$

It was suggested in [60] to replace the program (26) by

$$\min_{(z,w)\in\mathbb{R}^{n+m}} \|z\|_1 \qquad \text{s.t.} \qquad \|V(Az + w) - Vq\|_2 \leq \eta, \ \|w\|_2 \leq \kappa. \tag{27}$$

The following result summarizes two reconstruction results for subgaussian [60] and randomly subsampled subgaussian circulant matrices [26].

Theorem 10 ([60, Theorem 9]) and [26, Theorem 5]) *Let Q be the stable r-th order $\Sigma\Delta$-scheme with the one-bit alphabet $\mathcal{Q} = \{-1, 1\}$ as above and let $C_{Q,\mu}$ be its stability constant. Let $A \in \mathbb{R}^{m\times n}$ be a subgaussian matrix. Suppose that*

$$m \geq p \geq Cs\log(en/s).$$

Then the following holds with probability at least $1 - e^{-cp}$. For any $x \in \mathbb{R}^n$ satisfying $\|Ax\|_\infty \leq \mu < 1$ and $q = Q(Ax + \nu)$ with $\|\nu\|_\infty \leq \varepsilon < 1 - \mu$, any solution $x^\#$ to (26) *with $V = D^{-r}$, $\eta = C_{Q,\mu}\sqrt{m}$, $\kappa = \varepsilon\sqrt{m}$ satisfies*

$$\|x^\# - x\|_2 \lesssim_{\mu,r} \left(\frac{p}{m}\right)^{r-\frac{1}{2}} + \frac{\sigma_s(x)_1}{\sqrt{s}} + \sqrt{\frac{m}{p}}\varepsilon, \tag{28}$$

where $\sigma_s(x)_1 = \min_{z\in\Sigma_s} \|x - z\|_1$.

If A is a randomly subsampled subgaussian circulant matrix (with rows selected according to the uniformly at random model), then the same result holds with probability at least $1 - e^{-t}$ provided that, for some $0 \leq \alpha < 1/2$,

$$m \gtrsim t^{1/(1-2\alpha)} s \log^{2/(1-2\alpha)}(s) \log^{2/(1-2\alpha)}(n)$$

and $p = m(\frac{s}{m})^\alpha$.

The result in Theorem 10 essentially relies on proving that the matrix $D^{-r}A$ satisfies $\mathrm{RIP}_{2,2}(\Sigma_s, c)$, which has proven to be difficult for structured random matrices. To overcome this problem, [36] constructed a different preconditioner V for $\Sigma\Delta$-schemes as follows. For $p < m$ let $\lambda = m/p$. For simplicity, we assume that $\lambda \in \mathbb{N}$ and that there is a $\tilde{\lambda} \in \mathbb{N}$ such that $\lambda = r\tilde{\lambda} - r + 1$. Suppose that $u \in \mathbb{R}^\lambda$ contains the coefficients of the polynomial $(1 + z + \ldots + z^{\tilde{\lambda}-1})^r$. Define $V \in \mathbb{R}^{p\times m}$ by

$$V_{\Sigma\Delta} = \frac{1}{\sqrt{p}\|u\|_2} I_p \otimes u^T = \frac{1}{\sqrt{p}\|u\|_2} \begin{bmatrix} u^T & 0 & \cdots & 0 \\ 0 & u^T & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & u^T \end{bmatrix}. \tag{29}$$

Using this construction, [36] obtained the following result for partial Bernoulli circulant matrices with randomized row signs. It can be easily modified in the case of pre-quantization noise to produce an error bound similar to (28). In addition, a similar result was obtained for randomly subsampled discrete bounded orthonormal systems (again with randomized row signs).

Theorem 11 ([36, Theorem 6.1]) *Let Q be the stable r-th order $\Sigma\Delta$-scheme with the one-bit alphabet $\mathcal{Q} = \{-1, 1\}$ as above. Let B be a partial Bernoulli circulant matrix (with rows selected according to the row picking model), let D_ξ be a diagonal* *matrix with i.i.d. symmetric Bernoulli random variables on its diagonal which are independent of* B *and let* $A = D_\xi B$*. Fix* $\theta > 0$*,* $s \in [n]$ *and suppose that*

$$m \geq p \geq Cs \log^4 n.$$

Then the following holds with probability at least $1 - e^{-cp^2/(sm)}$*. For any* $x \in \mathbb{R}^n$ *satisfying* $\|Ax\|_\infty \leq \mu < 1$ *and* $q = Q(Ax)$*, any solution to* (26) *with* $V = V_{\Sigma\Delta}$ *satisfies*

$$\|x^\# - x\|_2 \lesssim_{\mu,r} \left(\frac{p}{m}\right)^{r-\frac{1}{2}} + \frac{\sigma_s(x)_1}{\sqrt{s}}.$$

The reconstruction error in Theorems 10 and 11 decays polynomially in terms of the number of measurements. If x is s-sparse ($\sigma_s(x)_1 = 0$) and there is no pre-quantization noise ($\varepsilon = 0$), then one can optimize the bound (28) (including the implicit constant depending on r) in terms of r. This yields an r depending on s and m for which the reconstructions error decays root-exponentially, i.e., as $e^{-\sqrt{m}}$, in terms of the number of measurements (see e.g., [60, Corollary 11]). Exponential error decay can be achieved by using a different noise-shaping method, called distributed noise-shaping quantization [11, 12]. For such recovery results with partial Bernoulli circulant matrices and randomly subsampled discrete bounded orthonormal systems (both with randomized row signs), see [36].

Acknowledgements It is a pleasure to thank the anonymous reviewer, Rayan Saab, and especially Laurent Jacques for many comments that improved this book chapter. This work was supported by the DFG through the project Quantized Compressive Spectrum Sensing (QuaCoSS), which is part of the Priority Program SPP 1798 Compressive Sensing in Information Processing (COSIP).

References

1. A. Ai, A. Lapanowski, Y. Plan, R. Vershynin, One-bit compressed sensing with non-Gaussian measurements. Linear Algebr. Appl. **441**, 222–239 (2014)
2. U. Ayaz, S. Dirksen, H. Rauhut, Uniform recovery of fusion frame structured sparse signals. Appl. Comput. Harmon. Anal. **41**(2), 341–361 (2016)
3. R. Baraniuk, S. Foucart, D. Needell, Y. Plan, M. Wootters, One-bit compressive sensing of dictionary-sparse signals. Inf. Inference: A J. IMA **7**(1), 83–104 (2017)
4. R.G. Baraniuk, S. Foucart, D. Needell, Y. Plan, M. Wootters, Exponential decay of reconstruction error from binary measurements of sparse signals. IEEE Trans. Inf. Theory **63**(6), 3368–3385 (2017)
5. P.T. Boufounos, R.G. Baraniuk, 1-bit compressive sensing, in *2008 42nd Annual Conference on Information Sciences and Systems* (IEEE 2008), pp. 16–21
6. P. T. Boufounos, L. Jacques, F. Krahmer, R. Saab, Quantization and compressive sensing, in *Compressed Sensing and its Applications* (Springer, 2015), pp. 193–237

7. J. Bourgain, An improved estimate in the restricted isometry problem, in *Geometric Aspects of Functional Analysis*, ed. B. Klartag, E. Milman, volume 2116 of *Lecture Notes in Mathematics* (Springer International Publishing, 2014), pp. 65–70
8. E.J. Candès, J., T. Tao, J.K. Romberg, Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. IEEE Trans. Inform. Theory **52**(2), 489–509 (2006)
9. E.J. Candès, J.K. Romberg, T. Tao, Stable signal recovery from incomplete and inaccurate measurements. Comm. Pure Appl. Math. **59**(8), 1207–1223 (2006)
10. E. Chou, *Beta-duals of frames and applications to problems in quantization*. PhD thesis, New York University (2013)
11. E. Chou, C.S. Güntürk, Distributed noise-shaping quantization: I. Beta duals of finite frames and near-optimal quantization of random measurements. Constr. Approx. **44**(1), 1–22 (2016)
12. E. Chou, C. S. Güntürk, Distributed noise-shaping quantization: II. Classical frames, in *Excursions in Harmonic Analysis, Volume 5* (Springer, 2017), pp. 179–198
13. E. Chou, C. S. Güntürk, F. Krahmer, R. Saab, Ö. Yılmaz, Noise-shaping quantization methods for frame-based and compressive sampling systems, in *Sampling Theory, a Renaissance* (Springer, 2015), pp. 157–184
14. I. Daubechies, R. DeVore, Approximating a bandlimited function using very coarsely quantized data: A family of stable sigma-delta modulators of arbitrary order. Ann. Math. **158**(2), 679–710 (2003)
15. M.A. Davenport, J. Romberg, An overview of low-rank matrix recovery from incomplete observations. IEEE J. Sel. Top. Signal Process. **10**(4), 608–622 (2016)
16. P. Deift, F. Krahmer, C.S. Güntürk, An optimal family of exponentially accurate one-bit sigma-delta quantization schemes. Commun. Pure Appl. Math. **64**(7), 883–919 (2011)
17. S. Dirksen, Dimensionality reduction with subgaussian matrices: a unified theory. Found. Comput. Math. **16**(5), 1367–1396 (2016)
18. S. Dirksen, H.C. Jung, H. Rauhut, One-bit compressed sensing with Gaussian circulant matrices. arXiv:1710.03287 (2017)
19. S. Dirksen, G. Lecué, H. Rauhut, On the gap between restricted isometry properties and sparse recovery conditions. IEEE Trans. Inform. Theory **64**(8), 5478–5487 (2018)
20. S. Dirksen, S. Mendelson, Non-gaussian hyperplane tessellations and robust one-bit compressed sensing. arXiv:1805.09409
21. S. Dirksen, S. Mendelson, Robust one-bit compressed sensing with partial circulant matrices. arXiv:1812.06719
22. S. Dirksen, S. Mendelson. Unpublished manuscript
23. D.L. Donoho, Compressed sensing. IEEE Trans. Inform. Theory **52**(4), 1289–1306 (2006)
24. A. Eftekhari, M.B. Wakin, New analysis of manifold embeddings and signal recovery from compressive measurements. Appl. Comput. Harmon. Anal. **39**(1), 67–109 (2015)
25. J.-M. Feng, F. Krahmer, An RIP-based approach to $\Sigma\Delta$ quantization for compressed sensing. IEEE Signal Process. Lett. **21**(11), 1351–1355 (2014)
26. J.-M. Feng, F. Krahmer, R. Saab, Quantized compressed sensing for partial random circulant matrices. arXiv:1702.04711 (2017)
27. S. Foucart, *Flavors of Compressive Sensing* (Springer International Publishing, Cham, 2017), pp. 61–104
28. S. Foucart, R. Lynch, Recovering low-rank matrices from binary measurements. Preprint (2018)
29. S. Foucart, H. Rauhut, *A Mathematical Introduction to Compressive Sensing* (Applied and Numerical Harmonic Analysis. Birkhäuser/Springer, New York, 2013)
30. V.K. Goyal, M. Vetterli, N.T. Thao, Quantized overcomplete expansions in $\mathbb{R}^N$ analysis, synthesis, and algorithms. IEEE Trans. Inform. Theory **44**(1), 16–31 (1998)
31. R.M. Gray, D.L. Neuhoff, Quantization. IEEE Trans. Inf. Theory **44**(6), 2325–2383 (1998)
32. R.M. Gray, T.G. Stockham, Dithered quantizers. IEEE Trans. Inf. Theory **39**(3), 805–812 (1993)
33. C.S. Güntürk, One-bit sigma-delta quantization with exponential accuracy. Commun. Pure Appl. Math. **56**(11), 1608–1630 (2003)

34. C.S. Güntürk, M. Lammers, A.M. Powell, R. Saab, Ö. Yılmaz, Sobolev duals for random frames and $\Sigma\Delta$ quantization of compressed sensing measurements. Found. Comput. Math. **13**(1), 1–36 (2013)
35. I. Haviv, O. Regev, The restricted isometry property of subsampled Fourier matrices, in *SODA '16* (Philadelphia, PA, USA, 2016), pp. 288–297
36. T. Huynh, R. Saab, Fast binary embeddings, and quantized compressed sensing with structured matrices. arXiv:1801.08639 (2018)
37. L. Jacques, A quantized Johnson-Lindenstrauss lemma: the finding of Buffon's needle. IEEE Trans. Inf. Theory **61**(9), 5012–5027 (2015)
38. L. Jacques, Error decay of (almost) consistent signal estimations from quantized gaussian random projections. IEEE Trans. Inf. Theory **62**(8), 4696–4709 (2016)
39. L. Jacques, Small width, low distortions: quantized random embeddings of low-complexity sets. IEEE Trans. Inf. Theory **63**(9), 5477–5495 (2017)
40. L. Jacques, V. Cambareri, Time for dithering: fast and quantized random embeddings via the restricted isometry property. Inf. Inference: A J. IMA **6**(4), 441–476 (2017)
41. L. Jacques, K. Degraux, C. De Vleeschouwer, Quantized iterative hard thresholding: Bridging 1-bit and high-resolution quantized compressed sensing. arXiv:1305.1786 (2013)
42. L. Jacques, J.N. Laska, P.T. Boufounos, R.G. Baraniuk, Robust 1-bit compressive sensing via binary stable embeddings of sparse vectors. IEEE Trans. Inform. Theory **59**(4), 2082–2102 (2013)
43. K. Knudson, R. Saab, R. Ward, One-bit compressive sensing with norm estimation. IEEE Trans. Inform. Theory **62**(5), 2748–2758 (2016)
44. F. Krahmer, S. Mendelson, H. Rauhut, Suprema of chaos processes and the restricted isometry property. Comm. Pure Appl. Math. **67**(11), 1877–1904 (2014)
45. F. Krahmer, R. Saab, Ö. Yilmaz, Sigma-delta quantization of sub-gaussian frame expansions and its application to compressed sensing. Inf. Inference **3**(1), 40–58 (2014)
46. J.N. Laska, P.T. Boufounos, M.A. Davenport, R.G. Baraniuk, Democracy in action: quantization, saturation, and compressive sensing. Appl. Comput. Harmonic Anal. **31**(3), 429–443 (2011)
47. M. Lustig, D.L. Donoho, J.M. Santos, J.M. Pauly, Compressed sensing MRI. IEEE Signal Process. Mag. **25**(2), 72–82 (2008)
48. S. Mendelson, Learning without concentration. J. ACM **62**(3), Art. 21, 25 (2015)
49. S. Mendelson, H. Rauhut, R. Ward, Improved bounds for sparse recovery from subsampled random convolutions. Ann. Appl. Probab. **28**(6), 3491–3527 (2018)
50. A. Montanari, N. Sun, Spectral algorithms for tensor completion. Comm. Pure Appl. Math. **71**(11), 2381–2425 (2018)
51. A. Moshtaghpour, L. Jacques, V. Cambareri, K. Degraux, C. De Vleeschouwer, Consistent basis pursuit for signal and matrix estimates in quantized compressed sensing. IEEE Signal Process. Lett. **23**(1), 25–29 (2016)
52. S. Oymak, B. Recht, Near-optimal bounds for binary embeddings of arbitrary sets. arXiv:1512.04433 (2015)
53. Y. Plan, R. Vershynin, One-bit compressed sensing by linear programming. Comm. Pure Appl. Math. **66**(8), 1275–1297 (2013)
54. Y. Plan, R. Vershynin, Robust 1-bit compressed sensing and sparse logistic regression: a convex programming approach. IEEE Trans. Inform. Theory **59**(1), 482–494 (2013)
55. Y. Plan, R. Vershynin, Dimension reduction by random hyperplane tessellations. Discrete Comput. Geom. **51**(2), 438–461 (2014)
56. H. Rauhut, R. Schneider, Z. Stojanac, Low rank tensor recovery via iterative hard thresholding. Linear Algebra Appl. **523**, 220–262 (2017)
57. L. Roberts, Picture coding using pseudo-random noise. IRE Trans. Inf. Theory **8**(2), 145–154 (1962)
58. J. Romberg, Compressive sensing by random convolution. SIAM J. Imaging Sci. **2**(4), 1098–1128 (2009)

59. M. Rudelson, R. Vershynin, On sparse reconstruction from Fourier and Gaussian measurements. Comm. Pure Appl. Math. **61**(8), 1025–1045 (2008)
60. R. Saab, R. Wang, Ö. Yılmaz, Quantization of compressive samples with stable and robust recovery. Appl. Comput. Harmonic Anal. **44**(1), 123–143 (2018)
61. G. Schechtman, Two observations regarding embedding subsets of Euclidean spaces in normed spaces. Adv. Math. **200**(1), 125–135 (2006)
62. R. Vershynin, *High-Dimensional Probability* (Cambridge University Press, 2018)
63. C. Xu, L. Jacques, Quantized compressive sensing with RIP matrices: the benefit of dithering. arXiv:1801.05870 (2018)

On Reconstructing Functions from Binary Measurements

Robert Calderbank, Anders Hansen, Bogdan Roman and Laura Thesing

Abstract We consider the problem of reconstructing a function from linear binary measurements. That is, the samples of the function are given by inner products with functions taking only the values 0 and 1. We consider three particular methods for this problem, the parameterized-background data-weak (PBDW) method, generalized sampling and infinite-dimensional compressed sensing. The first two methods are dependent on knowing the stable sampling rate when considering samples by Walsh function and wavelet reconstruction. We establish linearity of the stable sampling rate, which is sharp, allowing for optimal use of these methods. In addition, we provide recovery guaranties for infinite-dimensional compressed sensing with Walsh functions and wavelets.

R. Calderbank
Department of Electrical and Computer Engineering, Duke University, Durham, NC, USA
e-mail: robert.calderbank@duke.edu

A. Hansen · L. Thesing
Department of Applied Mathematics and Theoretical Physics, University of Cambridge, Cambridge, UK
e-mail: ach70@cam.ac.uk

L. Thesing
e-mail: lt420@damtp.cam.ac.uk

B. Roman (✉)
Department of Pure Mathematics and Mathematical Statistics, University of Cambridge, Cambridge, UK
e-mail: abr28@cam.ac.uk

H. Boche et al. (eds.), *Compressed Sensing and Its Applications*,
Applied and Numerical Harmonic Analysis,
https://doi.org/10.1007/978-3-319-73074-5_3

1 Introduction

A classical problem in sampling theory is to reconstruct a function f, that is typically in $L^2([0, 1]^d)$, from linear measurements in the form of inner products. One of the most famous problems of this type is to reconstruct f from its Fourier coefficients, where one can view the Fourier coefficients as values obtained from inner products of f with the basis of complex exponentials. In a more general abstract form, the problem is as follows.

Problem 1 An element $f \in \mathcal{H}$, where $\mathcal{H}$ is a separable Hilbert space, is to be reconstructed from measurements with linear functionals $(l_i)_{i\in\mathbb{N}} : \mathcal{H} \to \mathbb{C}$ that can be represented by elements $\zeta_i \in \mathcal{H}$ as $l_i(f) = \langle f, \zeta_i\rangle$. The key issue is that the l_i cannot be chosen freely, but are dictated by the modality of the sampling device.

Classical Fourier sampling problems in applications include magnetic resonance imaging (MRI), radio interferometry, etc., which is a natural consequence of the frequent appearance of the Fourier transform in the sciences. However, there is another important form of measurements, namely binary sampling. By binary sampling, we mean sampling with inner products of functions $\{\zeta_i\}_{i\in\mathbb{N}} \subset L^2([0, 1]^d)$ that only take the values 0 and 1. Just as Fourier sampling occurs naturally in many sampling devices, binary sampling is a phenomenon that occurs as a consequence of a sampling apparatus being ‘on’ or ‘off’, which occurs in digital signal processing, such as $\Sigma\Delta$ quantization, or newer forms of compressed measurements in microscopy or imaging.

There is a standard trick to convert binary sampling to measurements to functions that take the values $\{-1, 1\}$ rather than $\{0, 1\}$, by multiplying every measurement by 2 and subtracting the sample done with the constant function. Thus, one may assume, if one is willing to accept a potential change in the statistical noise model, that the measurements are done with $\{-1, 1\}$ valued functions. One motivation for converting from $\{0, 1\}$ valued sampling to $\{-1, 1\}$ is that the latter allows for the use of Walsh functions. These functions have very similar qualities to the complex exponentials in Fourier, and the Walsh transform is a close cousin of the Fourier transform. Moreover, the classical discrete Fourier transform obeys a fast implementation. This is also existent for the Walsh case via the Hadamard transform.

Given the extensive theory of Walsh functions and the benefits listed above, we will from now on assume that the sampling functions $\{\zeta_i\}_{i\in\mathbb{N}} \subset L^2([0, 1]^d)$ are the Walsh functions. A bonus property of the Walsh functions is that when combined with wavelets φ_j, $j \in \mathbb{N}$, spanning $L^2([0, 1]^d)$, in a change of basis matrix

$$U = \{\langle\varphi_j, \zeta_i\rangle\}_{i,j\in\mathbb{N}}, \tag{1}$$

one obtains a very structured infinite matrix. This infinite matrix shares many structural similarities with the change of basis matrix obtained by combining complex exponentials and wavelets. In particular, both types of infinite matrices become almost block diagonal, a feature that will be highly useful as we will see below.

In this paper, we will address three methods for Problem 1, two linear and one non-linear method. We choose the two linear ones because of their optimality with respect to the reconstruction error. The first is optimal in the class of linear methods that are consistent with the measurement and the second is optimal for linear algorithms that map into the reconstruction space. The non-linear method is the algorithm which takes the most structure into account and hence allows very good reconstruction guarantees. In all cases, we assume sampling with Walsh functions. The methods are as follows:

(i) The parameterized-background data-weak (PBDW) method (linear),
(ii) Generalized sampling (linear),
(iii) Infinite-dimensional compressed sensing (non-linear).

The PBDW method originated with the work by Maday, Patera, Penn and Yano in [51], and was further developed and analysed by by Binev, Cohen, Dahmen, DeVore, Petrova, and Wojtaszczyk [10, 13, 22]. Generalized sampling has been studied by Adcock, Hansen, Hrycak, Gröchenig, Kutyniok, Ma, Poon, Shadrin and others [1, 2, 4, 6, 40, 43, 50], and the predecessor; consistent sampling, has been analysed by Aldroubi, Eldar, Unser and others [8, 29–32, 64]. Infinite-dimensional compressed sensing has been developed and studied by Adcock, Hansen, Kutyniok, Lim, Poon and Roman [5, 7, 47, 55].

The successful use of the two first methods when reconstructing in a wavelet basis is completely dependent on the stable sampling rate which is defined below in terms of subspace angles between sampling and reconstruction spaces. It dictates the size of the sampling space as a function of the dimension of the reconstruction space in order to ensure accurate reconstructions. The main question we want to answer is

What is the stable sampling rate when given Walsh samples and a wavelet reconstruction basis?

The key issue is that the error bounds for these methods depend (sharply) on the subspace angle, and fortunately, we can provide sharp results on the stable sampling rate. In the case of infinite-dimensional compressed sensing, one cannot directly use the stable sampling rate, however, we provide estimates on the size of the sampling space as well as recovery guaranties from subsampled data.

To define the stable sampling rate we need to introduce some notation. The goal is to reconstruct f from the finite number of samples $\{l_i(f)\}_{i=1}^M$ for some $M \in \mathbb{N}$. The space of the functions ζ_i is called the *sampling space* and is denoted by $\mathcal{S} = \overline{\operatorname{span}}\{\zeta_i : i \in \mathbb{N}\}$, meaning the closure of the span. In practice, one can only acquire a finite number of samples. Therefore, we denote by $\mathcal{S}_M = \operatorname{span}\{\zeta_i : i = 1, \dots, M\}$ the sampling space of the first M elements. The reconstruction is typically done via a reconstruction space denoted by $\mathcal{R}$ and spanned by reconstruction functions $(\varphi_i)_{i\in\mathbb{N}}$, i.e. $\mathcal{R} = \overline{\operatorname{span}}\{\varphi_i : i \in \mathbb{N}\}$. As in the case of the sampling space, it is impossible to acquire and save an infinite number of reconstruction coefficients. Hence, one has to restrict to a finite reconstruction space, which is denoted by $\mathcal{R}_N = \operatorname{span}\{\varphi_i : i = 1, \dots, N\}$.

One of the main questions in reconstruction theory is how many samples are needed to guarantee a stable and accurate recovery? In the first part of this paper, we want to analyse this question for linear methods. The *stable sampling rate* captures the number of samples necessary to obtain a stable and accurate reconstruction of a certain number of coefficients in the reconstruction space. More precisely, we are interested in the dimension of the sampling space $\mathcal{S}_M$ in relation to the reconstruction space $\mathcal{R}_N$. Section 4 talks about the linear reconstruction method, and there we see that the accuracy and stability of the methods depend on the subspace angle between $\mathcal{R}_N$ and $\mathcal{S}_M$. In particular,

$$\cos(\omega(\mathcal{R}_N, \mathcal{S}_M)) := \inf_{r \in \mathcal{R}_N, \|r\|=1} \|P_{\mathcal{S}_M} r\|, \tag{2}$$

where $P_{\mathcal{S}_M}$ is the orthogonal projection onto the sampling space. Mainly, one is interested in the reciprocal value

$$\sigma(\mathcal{R}_N, \mathcal{S}_M) = 1/\cos(\omega(\mathcal{R}_N, \mathcal{S}_M)) \in [1, \infty], \tag{3}$$

which, as we will see later, plays a key role in all the error estimates of the two linear algorithms discussed here. Due to the definition of cosine, σ takes values in $[1, \infty]$. The stable sampling rate is then given by

$$\Theta(N, \theta) = \min \{M \in \mathbb{N} : \sigma(\mathcal{R}_N, \mathcal{S}_M) \leq \theta\}. \tag{4}$$

This function was analysed for different reconstruction methods in the Fourier case. We know that the stable sampling rate is linear for Fourier-wavelet [6] and Fourier-shearlet [50] reconstructions. This is the best one can wish for as it allows to reconstruct nearly as well from Fourier samples as sampling directly in the wavelet or shearlet system. However, this is not always the case. In the Fourier polynomial reconstruction, we get that the stable sampling rate is polynomial which leads to a large number of necessary samples and makes this only feasible for very sparse signals. For the Walsh case, it was shown in [41] that the stable sampling rate is also linear in the Walsh-wavelet case and it is even possible to determine the slope for Walsh–Haar wavelets [60].

The analysis for the non-linear reconstruction is a bit more involved and needs a detailed analysis of the change of basis matrix as well as the reconstruction space. It is common to use the sparsity of the coefficients in the wavelet space of natural images. However, it is known that this can be described more detailed with structured sparsity. This reduces the size of the class and hence allows better reconstruction guarantees. Similarly, we have that the change of basis matrix is not incoherent but asymptotically incoherent. This leads to a new version of CS with highly improved reconstruction quality from fewer samples. We will see that the impact of elements outside the diagonal boxes decays exponentially. This allows us to use more subsampling than previously described by the classical CS literature. We are here as well mainly interested in the reconstruction from Walsh samples with wavelets.

This paper is structured as follows. First, we discuss in Sect. 2 the Walsh functions, which are the building blocks of the sampling space. Then we revise the basics about boundary corrected wavelets in Sect. 3. With this information at hand, we are able to present the linear reconstruction methods and their analysis in Sect. 4. We then continue with the non-linear change of basis matrix including the classical theory, different problems type and then a detailed comparison between the new theory and the older versions. This is underlined by some numerical examples.

2 Walsh Functions

In this section, we introduce Walsh functions, which span the sampling space $\mathcal{S}_M$. First, we want to discuss the use of multi-indices. This is important because we want to deal with one- and d-dimensional functions. A multi-index j is commonly defined by $j = (j_1, \ldots, j_d) \in \mathbb{N}^d, d \in \mathbb{N}$. The basic operations such as addition and multiplication are understood pointwise, i.e. $j \star r = (j_1 \star r_1, \ldots, j_d \star r_d)$. This can also be done with a natural number n which is then interpreted as a multi-index with the same entry $n = (n, \ldots, n)$. Finally, the sum over a vector indexed by a multi-index is given by

$$\sum_{j=k}^{r} x_j := \sum_{j_1=k_1}^{r_1} \cdots \sum_{j_d=k_d}^{r_d} x_{j_1,\ldots,j_d}, \tag{5}$$

where $k, r \in \mathbb{N}^d$.

The multi-indices can be used to define functions in higher dimensions by the tensor product of the one-dimensional functions. The tensor product of $f : \mathbb{R} \to \mathbb{R}$ is given by

$$f(x) = f(x_1) \cdot \ldots \cdot f(x_d), \tag{6}$$

where $\{x_i\}_{i=1,\ldots,d} = x \in \mathbb{R}^d$ with $x_i \in \mathbb{R}$. Hence, the input parameter defines the dimensions. This simplifies the transition between the one- and d-dimensional case.

2.1 Definition

It is important to notice that Walsh functions behave very similarly to the complex exponential functions when the setting is changed from the decimal to the dyadic analysis. Dyadic analysis is a framework where functions are analysed for the situation where decimal addition is replaced by dyadic addition. Therefore, we start with a short review of the dyadic representation and addition. Let $x \in \mathbb{R}_+$, the dyadic representation is given by

$$x = \sum_{i \in \mathbb{Z}} x_i 2^i, \tag{7}$$

where $x_i \in \{0, 1\}$ for all $i \in \mathbb{Z}$. To make this representation unique, we use the one that ends in 0 instead of 1 if there is a choice. The dyadic addition of two numbers $x, y \in \mathbb{R}_+$ is given by

$$x \oplus y = \sum_{i \in \mathbb{Z}} (x_i \oplus_2 y_i) 2^i, \tag{8}$$

where $x_i \oplus_2 y_i$ is addition modulo two, i.e. $0 \oplus_2 0 = 0, 0 \oplus_2 1 = 1, 1 \oplus_2 0 = 1, 1 \oplus_2 1 = 0$. This definition can also be extended to all $\mathbb{R}$ and works as in the decimal case. We use the convention to denote negative numbers with a $-$.

With this information at hand, we can now define the Walsh functions, which span the sampling space $\mathcal{S}_M$.

Definition 1 ([36]) Let $t \in \mathbb{N}$ and $x \in [0, 1)$ with the dyadic representation $(t_0, \ldots)$ and $(\ldots, x_{-1})$. Then there exists a unique $n = n(t) \in \mathbb{N}$ such that $t = \sum_{i=0}^{n-1} t_i 2^i$, in particular $t_n \neq 0$ and $t_k = 0$ for all $k \geq n$. Let $t^n = (t_0, \ldots, t_n)$ and for $x = \sum_{i=-\infty}^{-1} x_i 2^i$ define $x^n = (x_{-n}, \ldots, x_{-1})$, and $C_W \colon \mathbb{R}^n \mapsto \mathbb{R}^n$ by

$$C_W = \begin{pmatrix} 0 \cdots & 0 & 1 & 1 \\ \vdots \;\; \iddots & \iddots & 1 & 0 \\ 0 \;\; \iddots & \iddots & \iddots & \vdots \\ 1 \;\;\; 1 & \iddots & & \vdots \\ 1 \;\;\; 0 & \cdots & \cdots & 0 \end{pmatrix}. \tag{9}$$

The *Walsh functions* are then given by

$$\mathrm{wal}(t; x) = (-1)^{t^n \cdot C_W x^n}. \tag{10}$$

This definition is a bit longer to write, however, it gives an interesting insight into the ordering of the Walsh functions. There are a lot of different orderings of the Walsh functions available. The first choice for the matrix C_W might be the identity. The functions are then called Walsh–Kronecker functions. The problem with this ordering is that the functions change completely if the maximal element $n(t)$ is changed. Therefore, they are seldom used in practice. One attempt to overcome this problem is the Walsh–Paley ordering which is given by the reversal matrix:

$$C_{WP} = \begin{pmatrix} 0 \cdots & 0 & 0 & 1 \\ \vdots \;\; \iddots & \iddots & 1 & 0 \\ 0 \;\; \iddots & \iddots & \iddots & 0 \\ 0 \;\;\; 1 & \iddots & \iddots & \vdots \\ 1 \;\;\; 0 & 0 & \cdots & 0 \end{pmatrix}. \tag{11}$$

In this scenario, the functions stay the same with changing $n(t)$. Hence, they overcome one drawback of the Walsh–Kronecker ordering. However, they are not ordered with increasing number of zero crossing. This would be a desirable property because, first, it makes them similar to exponential functions and second, it relates well to the level ordering of the wavelets. Therefore, we use the presented definition as it obeys none of the discussed drawbacks.

It is also possible to extend the classical Walsh functions to inputs in $\mathbb{R}_+ \times \mathbb{R}_+$, i.e.

$$\mathrm{Wal}(t, x) = (-1)^{t_0 x_0}\, \mathrm{wal}([t]\,; x)\, \mathrm{wal}([x]\,; t), \tag{12}$$

where t and x have the dyadic representation $(t_i)_{i\in\mathbb{Z}}$ and $(x_i)_{i\in\mathbb{Z}}$, t_0, x_0 are the corresponding elements of the sequence and $[\cdot]$ denotes the rounding down operation. We get the same functions if C_W is defined over $\mathbb{N}$ instead of $\{1, \dots, n(t)\}$. For negative inputs, we take the same definition as in [36]

$$\mathrm{Wal}(-t, x) := -\,\mathrm{Wal}(t, x) \tag{13}$$

$$\mathrm{Wal}(t, -x) := -\,\mathrm{Wal}(t, x). \tag{14}$$

With the presentation of the multi-indices and the generalized Walsh functions, it is now easy to define them in higher dimensions with the tensor product by

$$\mathrm{Wal}(t, x) = \bigotimes_{k=1}^{d} \mathrm{Wal}(t_k, x_k), \tag{15}$$

where $t = \{t_k\}_{k=1,\dots,d}$, $x = \{x_k\}_{k=1,\dots,d} \in \mathbb{R}^d$. These function then span the sampling space, i.e.

$$\mathcal{S} = \overline{\mathrm{span}} \left\{\mathrm{Wal}(t, \cdot), t \in \mathbb{N}^d\right\} \subset L^2([0, 1]^d) \tag{16}$$

and for $M = m^d$ for some $m \in \mathbb{N}$ we have

$$\mathcal{S}_M = \mathrm{span}\,\{\mathrm{Wal}(t, \cdot), t_i \le m, i = 1, \dots, d\} \subset L^2([0, 1]^d) \tag{17}$$

For discrete signals in $\mathbb{C}^N$, the orthogonal projection onto the sampling space is often denoted by Ψ, which we shall discuss later on.

Finally, we define a continuous and a discrete transform. For the definition of the continuous transform, we have to ensure that the integral exists. Therefore, let $f \in L^2([0, 1]^d)$ then the *generalized Walsh transform* is given almost everywhere by

$$\widehat{f}^{W}(t) = \mathcal{W}\{f(\cdot)\}(t) = \langle f(\cdot), \mathrm{Wal}(t, \cdot)\rangle = \int_{[0,1]^d} f(x)\, \mathrm{Wal}(t, x) dx, \quad t \in \mathbb{R}^d. \tag{18}$$

The restrictions to functions which are supported in $[0, 1]^d$ leads to the use of boundary corrected wavelets which are presented in Sect. 3. For the discrete transform, let $N = 2^n$, $n \in \mathbb{N}$ and $x = \{x_0, \ldots, x_{N-1}\} \in \mathbb{R}^N$ then the one-dimensional *discrete Walsh transform* of x is given by $X = \{X_0, \ldots, X_{N-1}\}$ with

$$X_j = \frac{1}{N} \sum_{k=0}^{N-1} x_k \operatorname{Wal}\big(j, \frac{k}{N}\big). \tag{19}$$

As discussed before, the Walsh functions are desirable because of the fast transform. It can be seen here, that this indeed corresponds to the Hadamard transform and therefore, the Walsh functions are its kernel.

In higher dimensions, we get for $x \in \mathbb{R}^{N_1 \times \ldots \times N_d}$ where $x_{k_i} \in \mathbb{R}$, $k = \{k_i\}_{i=1,\ldots,d}$, $k_i = 0, \ldots, N_i - 1$ the discrete Walsh transformed $X = \{X_j\} \in \mathbb{R}^{N_1 \times \ldots \times N_d}$, where $X_{j_i} \in \mathbb{R}$, $j = \{j_i\}_{i=1,\ldots,d}$, $j_i = 0, \ldots, N_i - 1$, with

$$X_j = \frac{1}{\prod_{i=1}^d N_i} \sum_{k=0}^{N-1} x_k \operatorname{Wal}(j, \frac{k}{N}). \tag{20}$$

2.2 Properties

In this section, we recall the most important and useful properties of the Walsh functions and transfer them to the continuous transform. The Walsh functions are symmetric,

$$\operatorname{Wal}(t, x) = \operatorname{Wal}(x, t)\, \forall t, x \in \mathbb{R}, \tag{21}$$

and they obey the *scaling property* as well as the *multiplicative identity*, i.e.

$$\operatorname{Wal}(2^k t, x) = \operatorname{Wal}(t, 2^k x)\, \forall t, x \in \mathbb{R},\ k \in \mathbb{N} \tag{22}$$

and

$$\operatorname{Wal}(t, x) \operatorname{Wal}(t, y) = \operatorname{Wal}(t, x \oplus y)\, \forall t, x, y \in \mathbb{R}. \tag{23}$$

Due to the tensor product definition, these properties also hold in the d-dimensional case. Moreover, we have for the transform, that it is linear:

$$\mathcal{W}\{af(x) + bg(x)\} = a\mathcal{W}\{f(x)\} + b\mathcal{W}\{g(x)\}\ \forall a, b \in \mathbb{R},\ f, g \in L^2([0,1]^d), \tag{24}$$

obeys the following *shift* and *scaling property*, i.e.

$$\mathcal{W}\{f(x \oplus y)\}(t) = \mathcal{W}\{f(y)\}(t) \operatorname{Wal}(x, t)\, \forall x \in \mathbb{R}^d,\ f \in L^2([0,1]^d) \tag{25}$$

and

$$\mathcal{W}\left\{f(2^m x)\right\}(t) = \frac{1}{2^m}\mathcal{W}\{f(x)\}(\frac{t}{2^m})\, \forall m \in \mathbb{N}^d,\ f \in L^2([0,1]^d). \tag{26}$$

3 Reconstruction Space

The reconstruction space should be chosen appropriately for the given data. For image and audio signals, wavelets have proven to be very useful as they are able to present the data sparsely. In the following, we will deal with Daubechies wavelets. Normally, they are defined on the whole $\mathbb{R}^d$. However, we need to have that $\mathcal{S}_M^{\perp} \cap \mathcal{R}_N = \{0\}$ because otherwise there are elements in the reconstruction space which cannot be captured with the sampling space and makes it impossible to a have unique solution to the reconstruction problem. Hence, we have to restrict ourselves to wavelets that are only defined on the cube $[0, 1]^d$. For this sake, we use boundary corrected wavelets and in higher dimensions separable boundary corrected wavelets which are constructed by tensor products. We follow the construction as in [20].

For a smoother outline of boundary wavelets, we start with the one-dimensional case. We denote the mother wavelet with ψ and the corresponding scaling function with ϕ, which is equal to the common literature in this area. The corresponding wavelet and scaling spaces are spanned by the scaled and translated versions

$$\psi_{r,j}(x) := 2^{r/2}\psi(2^r x - j) \text{ and } \phi_{r,j}(x) := 2^{r/2}\phi(2^r x - j), \tag{27}$$

where $r, j \in \mathbb{Z}$. With this, we obtain the wavelet space $W_r := \text{span}\left\{\psi_{r,j} : j \in \mathbb{Z}\right\}$ at level r and accordingly, the scaling space $V_r := \text{span}\left\{\phi_{r,j} : j \in \mathbb{Z}\right\}$. As discussed at the beginning of the chapter and in the previous chapter we have to restrict ourselves to functions defined on $[0, 1]^d$. Therefore, we take boundary-corrected Daubechies wavelets which are introduced in Sect. 4 in [20]. They have two major advantages. The first is the maintained smoothness and compactness properties of the original wavelet. Second, they also keep the multi-resolution analysis. This is important for the definition of the higher dimensional wavelets. It allows us to keep the structure also in higher dimensions. We can still represent the reconstruction space with the scaling space in only one level.

We start with the scaling function at the level J_0 such that the functions can only intersect with one boundary at a time. The scaling space is then given by

$$V_{J_0}^b = \text{span}\left\{\phi_{J_0,j} : j = 0, \dots, 2^{J_0} - p - 1, \phi_{J_0,j}^{\#} : j = 2^{J_0} - p, \dots, 2^{J_0} - 1\right\}, \tag{28}$$

where $\phi^{\#}$ is the scaling function reflected at 1. The definition for higher levels $r > J_0$ works accordingly. We denote the boundary wavelets by ψ^b and $\psi_{j,m}^b(x) = 2^{j/2}\psi(2^j x - m)$ for $j \geq J_0$. We are only interested in the smoothness properties of the wavelet, which stay the same to the generating wavelet ψ. Therefore, we do not get into the details about the construction of the ψ^b. Interested readers should seek out for [20] for a detailed explanation. The wavelet space is then given by

$$W_r^b = \text{span}\left\{\psi_{r,j}^b : j = 0, \dots, 2^r\right\}. \tag{29}$$

For the linear reconstruction methods, it suffices to only consider the reconstruction space as a whole. Therefore, we exploit the multi-resolution analysis and we can represent the wavelet spaces up to level $R-1$ by the scaling space at level R, i.e.

$$\bigcup_{r<R} W_r^b = V_R^b. \tag{30}$$

This allows us for a number of coefficients related to the levels, i.e. $N = 2^R$ to represent the reconstruction space as

$$\mathcal{R}_N := V_R^b. \tag{31}$$

This has the positive byproduct that we do not have to deal with the internal ordering.

For the non-linear methods, this internal ordering becomes more important. We then let

$$\mathcal{R}_N := V_{J_0}^b \oplus W_{J_0}^b \ldots \oplus W_{R-1}^b. \tag{32}$$

We now get to the definition in higher dimensions. The scaling space is defined by the tensor product, i.e.

$$\mathcal{R}_N = V_R^{b,d} := V_R^b \otimes \ldots \otimes V_R^b \quad \text{(d-times)} \tag{33}$$

for $N = 2^{dR}$. It is important to note that the wavelet space in higher dimensions is not simply the tensor product of the one-dimensional wavelets, but the combination of wavelets and scaling functions, i.e.

$$V_j^{b,d} = V_j^b \otimes \ldots \otimes V_j^b = (V_{j-1}^b \oplus W_{j-1}^b) \otimes \ldots \otimes (V_{j-1}^b \oplus W_{j-1}^b) = V_{j-1}^{b,d} \oplus W_{j-1}^{b,d}. \tag{34}$$

And hence,

$$W_{j-1}^{b,d} := (V_{j-1}^b \oplus W_{j-1}^b) \otimes \ldots \otimes (V_{j-1}^b \oplus W_{j-1}^b) \ominus V_{j-1}^{b,d}. \tag{35}$$

Let $\phi_{J_0,m}^{b,d} = \bigotimes_{i=1}^d \phi_{J_0,m_i}^b$ and $\psi_{j,m}^{b,d} = \bigotimes_{i=1}^d \psi_{j,m_i}^b$, where ϕ^b can stand for ϕ or $\phi^{\#}$ depending on m. For the reconstruction space this results in

$$\mathcal{R} = \Big\{\phi_{J_0,m}^{b,d}, m = (m_1, \ldots, m_d), m_i = 0, \ldots, 2^{J_0} - 1 \tag{36}$$

$$\phi_{j,m}^{b,d-1} \otimes \psi_{j,m}, \ldots, \phi_{j,m}^b \otimes \psi_{j,m}^{b,d-1}, \psi_{j,m}^{b,d}, \tag{37}$$

$$j \geq J_0, m = (m_1, \ldots, m_d), m_i = 0, \ldots, 2^j - 1\Big\}. \tag{38}$$

Note the abuse of notation in $\phi_{j,m}^{b,d-r} \oplus \psi_{j,m}^{b,r}$. Only the parts of the multi-index $m = (m_1, \ldots, m_d)$ related to the position of the function in the tensor product are used for the shift of the function. Moreover, we have 2^d different possibilities to combine the scaling function and the wavelets by the tensor product. Hence, there are $2^{dj}(2^d - 1)$ elements at every level j. In case of doubt of the dimension, we will use an upper index d to make the distinction clear, i.e. for $u^d_{i,j}$. For the discrete setting the orthogonal projection onto the reconstruction space is often denoted by Φ, which will be discussed in more details with the numerical experiments.

4 Linear Reconstruction Methods

In this section, we are concerned with two different linear reconstruction methods the *PBDW method* and *generalized sampling*. They both have in common that they are linear and share the same condition number and that the accuracy is highly dependent on the stable sampling rate which is analysed in the last subsection.

4.1 PBDW Method

The PBDW method as introduced in [51] and analysed in [10, 13, 22] is based on the following idea. Given the measurements $l := P_{\mathcal{S}_M} f$, where $P_{\mathcal{S}_M}$ denotes the orthogonal projection onto the subspace $\mathcal{S}_M$, one tries to find an approximation that is consistent with the measurements and close to the reconstruction space $\mathcal{R}_N$, which is measured with the distance $\mathrm{dist}(f, \mathcal{R}_N) = \min\{||f - \varphi||_2 : \varphi \in \mathcal{R}_N\}$ and bounded by a sequence $\{\epsilon_N\}_{N\in\mathbb{N}}$. Mathematically, one tries to approximate f by $f^* \in \mathcal{K}_l$ where we define

$$\mathcal{K} = \{f \in \mathcal{H} : \mathrm{dist}(f, \mathcal{R}_N) \leq \epsilon_N\} \text{ and } \mathcal{H}_l = \left\{f \in \mathcal{H} : P_{\mathcal{S}_M} f = l\right\}, \tag{39}$$

and the space of possible approximation is then the intersection $\mathcal{K}_l := \mathcal{K} \cap \mathcal{H}_l$. Obviously, we try to find the closest element $f^* \in \mathcal{K}_l$ to the true solution f, hence, we solve the minimization problem

$$g^* = \mathrm{argmin}_{g\in\mathcal{R}_N} ||l - P_{\mathcal{S}_M} g||^2. \tag{40}$$

The outcome $g*$ is then adjusted to be consistent with the measurements by

$$f^* = l + P_{\mathcal{S}_M^\perp} g^*. \tag{41}$$

Then f^* is the solution to the PBDW method and was analysed in [13] and shown to be optimal with respect to the distance to the true function for all functions that are consistent with the measurements. We have the error estimate

$$||f - f^*|| \leq \sigma(\mathcal{R}_N, \mathcal{S}_M)\, \mathrm{dist}(f, \mathcal{R}_N). \tag{42}$$

This error estimate was then improved in [51] to

$$||f - f^*|| \leq \sigma(\mathcal{R}_N, \mathcal{S}_M)\,\mathrm{dist}(f, \mathcal{R}_N \oplus (\mathcal{S}_M \cap \mathcal{R}_N^{\perp})). \tag{43}$$

However, it was shown in [13] that the factor of the subspace angle $\sigma(\mathcal{R}_N, \mathcal{S}_M)$ cannot be removed or improved.

This underlines again that it is important to make sure that $\mathcal{R}_N \cap \mathcal{S}_M^{\perp} = \{0\}$. Moreover, it underlines the importance of the stable sampling rate and estimates of the relation between the number of samples M and the number of reconstructed coefficients N to get a stable and accurate reconstruction. In the next section, we discuss the concept of generalized sampling and see that the condition number of the PBDW method also equals the subspace angle, which underlines its importance.

4.2 Generalized Sampling

We now study a different linear reconstruction technique: *generalized sampling*. Unlike PBDW, it forces the solution to stay in the reconstruction space. In particular, for very sparse data in the reconstruction space, it improves the reconstruction quality.

The method is an extension of the finite section methods [14, 38, 39, 48]. In important cases like Fourier-wavelet or Walsh-wavelet the finite section method is very unstable. The advantage of generalized sampling is that it allows a different number of samples than reconstructed coefficients, which makes the method stable and accurate. The question of how to choose the number of measurements with respect to the number of coefficients is answered by the stable sampling rate. We give this method now, and then explain how it can be cast into a least squares problem.

Definition 2 ([1]) For $f \in \mathcal{H}$ and $N, M \in \mathbb{N}$ we define the reconstruction method of *generalized sampling* $G_{N,M} : \mathcal{H} \to \mathcal{R}_N$ by

$$\langle P_{\mathcal{S}_M} G_{N,M}(f), \varphi_i \rangle = \langle P_{\mathcal{S}_M} f, \varphi_i \rangle, \quad i = 1, \dots, N, \tag{44}$$

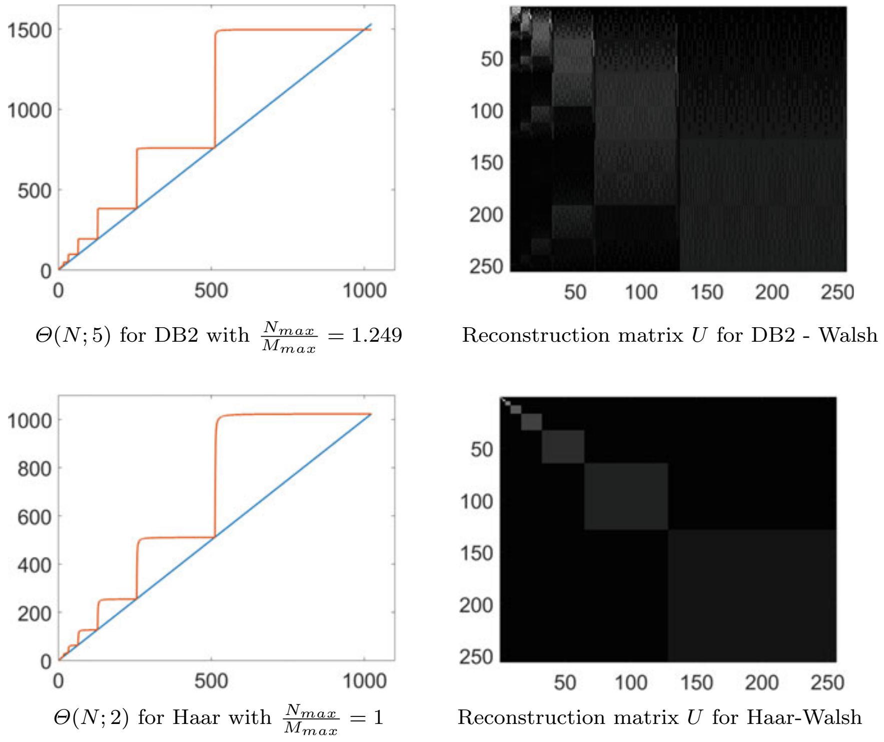
where $\varphi_i, i = 1, \dots, N$ span $\mathcal{R}_N$. We refer to $G_{N,M}(f)$ as the *generalized sampling reconstruction* of f.

Equation (44) can be rewritten as the following least squares problem: We search for a solution $\alpha^{[N]} \in \mathbb{R}^N$ of

$$U^{[N,M]} \alpha^{[N]} = l(f)^{[M]}, \tag{45}$$

where

$$U^{[N,M]} = \begin{pmatrix} u_{11} & \dots & u_{1N} \\ \vdots & \ddots & \vdots \\ u_{M1} & \dots & u_{MN} \end{pmatrix} \tag{46}$$



$\Theta(N;5)$ for DB2 with $\frac{N_{max}}{M_{max}} = 1.249$

Reconstruction matrix U for DB2 - Walsh

$\Theta(N;2)$ for Haar with $\frac{N_{max}}{M_{max}} = 1$

Reconstruction matrix U for Haar-Walsh

Fig. 1 Stable sampling rate and change of basis matrix for different wavelets with Walsh functions

and $u_{ij} = \langle \varphi_j, \zeta_i \rangle$, $l(f)^{[M]} = (l_1(f), \dots, l_M(f)) \in \mathbb{R}^M$. The solution of the method is then given by $G_{N,M}(f) = \sum_{i=1}^{N} \alpha_i \varphi_i$. The change of basis matrix U can be seen in Fig. 1 for Walsh measurements and different wavelets.

Generalized sampling was widely studied and it was shown that (44) yields a solution if the number of samples is large enough.

Theorem 1 ([2]) *Let $N \in \mathbb{N}$. Then, there exists $M_0 \in \mathbb{N}$ such that for every $f \in \mathcal{H}$ Eq.* (44) *has a unique solution $G_{N,M}(f)$ for all $M \geq M_0$. Moreover, the smallest M_0 is the least number such that* $\cos(\omega(\mathcal{R}_N, \mathcal{S}_{M_0})) > 0$.

Additionally, the condition number and optimality was analysed. Here it is interesting to notice, that the generalized sampling as well as the PBDW method are optimal in their setting and that in both cases the performance depends on the subspace angle between the sampling and reconstruction space $\sigma(\mathcal{R}_N, \mathcal{S}_M)$.

Theorem 2 ([2]) *Retaining the definitions and notations from this section, for all $f \in \mathcal{H}$ we have*

$$||G_{N,M}(f)|| \leq \sigma(\mathcal{R}_N, \mathcal{S}_M)||f||, \tag{47}$$

and

$$||f - P_{\mathcal{R}_N} f|| \leq ||f - G_{N,M}(f)|| \leq \sigma(\mathcal{R}_N, \mathcal{S}_M)||f - P_{\mathcal{R}_N} f||. \tag{48}$$

In particular, these bounds are sharp.

Remark that the same least square problem is solved for the PBDW method and generalized sampling. Therefore, the analysis of the condition number $\kappa(\mathcal{R}_N, \mathcal{S}_M)$ of the generalized sampling approach in [4] translates directly to the PBDW method. We get

$$\kappa(\mathcal{R}_N, \mathcal{S}_M) = \sigma(\mathcal{R}_N, \mathcal{S}_M). \tag{49}$$

Hence, the stable sampling rate is important to analyse the accuracy and also the stability.

4.3 The Stable Sampling Rate for the Walsh-Wavelet Case

In this section, we recall the results from [41] about the stable sampling rate for the Walsh-wavelet case.

Theorem 3 ([41]) *Let $\mathcal{S}$ and $\mathcal{R}$ be the sampling and reconstruction space spanned by the d-dimensional Walsh functions and separable boundary wavelets, respectively. Moreover, let $N = 2^{dR}$ with $R \in \mathbb{N}$. Then for all $\theta \in (1, \infty)$, there exists S_θ such that for all $M \geq 2^{dR} S_\theta$, we have $\sigma(\mathcal{R}_N, \mathcal{S}_M) \leq \theta$. In particular, one gets $\Theta \leq S_\theta N$. Hence, the relation $\Theta(N; \theta) = \mathcal{O}(N)$ holds for all $\theta \in (1, \infty)$.*

In Fig. 1, the stable sampling rate is displayed for different Daubechies wavelets. One can see that the slope S_θ is smaller for wavelets with a more block-diagonal change of basis matrix. A direct relation between the number of vanishing moments and the value of S_θ is not known due to the very different behaviour of Walsh functions and wavelets.

One should note that for the case of Haar wavelets, the slope $S_\theta = 1$ for all $\theta \in (1, \infty)$. This relation was analysed in more detail in [60].

Theorem 4 ([60]) *Let the sampling space $\mathcal{S}$ be spanned by the Walsh functions and the reconstruction space $\mathcal{R}$ by the Haar wavelets in $L^2([0, 1]^d)$. If $N = 2^{dR}$ for some $R \in \mathbb{N}$, then for every $\theta \in (1, \infty)$ we have that the stable sampling rate is the identity, i.e. $\Theta(N, \theta) = N$.*

These results show that sampling with Walsh functions is nearly as good as sampling directly with wavelets. Hence, the presented algorithms allow to improve the recovery quality. This can be seen in the comparison with direct inversion where one gets a lot of block artefacts from Walsh functions or the Gibbs phenomena with Fourier samples. These are mostly removed after the reconstruction method. We can analyse mathematically the approximation rate in the different bases.

4.3.1 Approximation Qualities

Approximation theory provides a useful tool for comparing the representation qualities of different bases. Let $\{\varphi_i\}_{i\in\mathbb{N}}$ be an orthonormal basis for $L^2([0,1])$, hence every $f \in L^2([0,1])$ can be represented by

$$f = \sum_{i\in\mathbb{N}} \langle f, \varphi_i\rangle \varphi_i. \tag{50}$$

In practice, this is not a feasible representation approach. This is due to the fact that we can only access and store a finite number of coefficients. Hence, instead of the true f we can only have an approximation $f_N = \sum_{i=1}^{N} \langle f, \varphi_i\rangle \varphi_i$. The resulting approximation error is given by

$$\epsilon(N, f) = ||f - f_N||_2^2 = \int |f - f_N|^2 dx = \sum_{i>N} |\langle f, \varphi_i\rangle|^2. \tag{51}$$

In approximation theory, one compares bases and representation systems in terms of the decay of $\epsilon(N, f)$ with respect to N for functions f in some specified function class. A very fast decay with N is desirable, because this allows a good representation from only a few coefficients, which then results in less measurements.

The decay rate of the Walsh transform of Lipschitz continuous functions is analysed in [9]. It was shown that

$$\langle f, \mathrm{Wal}(n, \cdot)\rangle = \widehat{W} f(n) \le 2^{-p}, \tag{52}$$

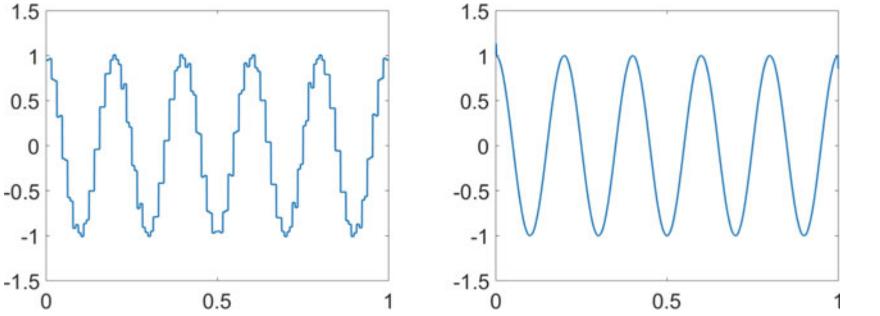
where $2^p \le n < 2^{p+1}$. With this we get for the approximation error

$$\epsilon(N, f) \le \sum_{i>N} \frac{1}{2i^2} \le \frac{1}{2N}$$

which then lies in $\mathcal{O}(N^{-1})$. In contrast to the Fourier transform this does not improve if the function gets smoother or periodic. The resulting artefacts can be seen in Fig. 2. Therefore, the reconstruction techniques as the PBDW method or generalized sampling are very useful because they allow to change the basis in which we represent our data. We then use a basis such as wavelets with a much faster decay rate. The decay rate is analysed in [52]. Daubechies wavelets of order p represent functions f in the Sobolev space $W^\gamma([0,1])$ for some $\gamma < p$ with an approximation rate of

$$\epsilon(N, f) = \mathcal{O}(N^{-2\gamma}). \tag{53}$$

This underlines the advantage of representing smooth functions with Daubechies wavelets instead of Walsh functions. Due to the findings in Theorem 3 it is possible to highly improve the reconstruction quality from binary measurements. In



Truncated Walsh series

Generalized sampling with 64 reconstructed wavelet coefficients

Fig. 2 Reconstruction from 77 binary measurements, where both examples use exactly the same samples. The figure illustrates the change in approximation rate when converting the Walsh samples to wavelet coefficients via generalized sampling

particular, because of the linearity of the stable sampling rate and a reasonable slow slope, for example $S_2 = 2$ for Daubechies wavelets of order 8, we get an improved representation from $\mathcal{O}(N^{-1})$ to $\mathcal{O}(N^{-2\gamma})$.

5 Non-linear Reconstruction Methods

In the previous section, we have seen linear reconstruction methods and discussed their convergence properties in view of subspace angles and the stable sampling rate. Even though they offer good results and fast computations, they are rather restrictive in terms of adaptivity to the problem at hand, in particular, they do not allow for subsampling. Hence, we want to extend our analysis to the non-linear methods where we focus on compressed sensing (CS) and the structure binary measurements provide that allows for substantial undersampling. The main issue with this extension is that classical compressed sensing considers finite-dimensional signals. However, we are dealing with infinite-dimensional ones. Hence, the classic compressed needs to be extended to infinite-dimensional compressed sensing introduced in [5, 7]. Nevertheless, we start this chapter with a quick review of the standard finite-dimensional compressed sensing.

5.1 *Classical Compressed Sensing*

Compressed sensing (CS) was introduced by Candès et al. [19] and Donoho [24] and is formulated in the finite-dimensional setting, stating that under appropriate conditions one can overcome the Nyquist sampling barrier and recover signals using far fewer samples than dictated by the classical Shannon theory.

A traditional CS setup is as follows. The aim is to recover a signal f from an incomplete (subsampled) set of measurements y. Here, f is represented as a vector in $\mathbb{C}^N$ and is assumed to be s-sparse in some orthonormal basis $\Phi \in \mathbb{C}^{N\times N}$ (e.g. wavelets) called the *reconstruction* or *sparsity* basis. This means that its vector of coefficients $x = \Phi f$ has at most s non-zero entries. Let $\Psi \in \mathbb{C}^{N\times N}$ be an orthonormal basis, called *sensing* or *sampling* basis, and write $U = \Psi\Phi^* = (u_{ij})$, which is an isometry and the discrete version of U in (46). The coherence of U is

$$\mu(U) = \max_{i,j} |u_{ij}|^2 \in [1/N, 1]. \tag{54}$$

and U is said to be perfectly incoherent if $\mu(U) = 1/N$.

Let the *subsampling pattern* be the set $\Omega \subseteq \{1, \dots, N\}$ of cardinality m with its elements chosen uniformly at random. This is one of the main differences to the previous discussed linear reconstruction methods. For the linear methods, we restrict ourselves to the first N measurements instead of choosing the most beneficial ones. Owing to a result by Candès and Plan [16] and Adcock and Hansen [5], if we have access to the subset of noisy measurements $y = P_\Omega \Psi f + e$ then f can be recovered from y exactly (up to the noise level) with probability at least $1 - \epsilon$ if

$$m \gtrsim \mu(U) \cdot N \cdot s \cdot (1 + \log(1/\epsilon)) \cdot \log(N), \tag{55}$$

where $P_\Omega \in \{0, 1\}^{N\times N}$ is the diagonal projection matrix with the j^{th} entry 1 if $j \in \Omega$ and 0 otherwise, and the notation $a \gtrsim b$ means that $a \geq C\, b$ where $C > 0$ is some constant independent of a and b. Then, f is recovered by solving

$$\min_{z\in\mathbb{C}^N} \|z\|_1 \quad \text{subject to} \quad \|y - P_\Omega U z\| \leq \eta. \tag{56}$$

where η is chosen according to the noise level, i.e. $\|e\| \leq \eta$. The key estimate (55) shows that the number of measurements m required is, up to a log factor, on the order of the sparsity s, provided the coherence $\mu(U) = \mathcal{O}(1/N)$. This is the case, for example, when U is the DFT, which was studied in some of the first CS papers [19].

The main reason why we want to consider infinite dimensional CS is threefold. First, our signal is defined in a continuous setting in $L([0, 1]^d)$ instead of $\mathbb{R}^n$, hence it is sensible to adapt the reconstruction problem accordingly. Second, the discrete setting leads to the measurement mismatch and wavelet crime. The measurement mismatch comes from the fact that the discrete Hadamard transform leads to an approximation of the signal by step function, which always results in an additional approximation error for our method. The wavelet crime describes the error which results from assuming that the discrete inverse of the wavelet coefficients leads to the point evaluations of the signal. This is an approximation, which uses the fact that for high order scaling function the support is nearly pointwise. However, as this is only an approximation we also add up this error. Finally, we want to analyse the change of basis matrix. For the analysis, it is easier to consider the inner products of the wavelets and the Walsh function than to work with the discrete matrices. This allows us to develop a rich analysis.

5.2 Types of Compressed Sensing Problems

CS problems can be roughly divided into two types. **Type I** are problems where the physical device imposes the sampling operator, but allows some limited freedom to design the sampling strategy. This category is vast, with examples including magnetic resonance imaging (MRI), electron microscopy (EM), computerized tomography, seismic tomography and radio interferometry. **Type II** are problems where the sensing mechanism offers freedom to design both the sampling operator and the strategy. Examples include fluorescence microscopy (FM) and compressive imaging (CI) (e.g. single pixel and lensless cameras). In these two examples, many practical setups still impose some restrictions regarding the sampling operator, e.g. measurements must typically be binary.

In a simplified view, traditional CS assumes three main principles: *sparsity* (there are s important coefficients in the vector to be recovered, however, the location is arbitrary), *incoherence* (the values in the measurements matrix should be uniformly spread out) and *sampling is performed with some degree of randomness*.

In many Type I practical problems, some of the above principles as introduced in the traditional CS model are lacking. For example, many Type I problems are coherent due to the physics of the underlying sensing mechanism. However, CS was used successfully in such problems, though with very different sampling techniques than uniform random subsampling. For Type II problems, the traditional CS framework is applicable, e.g. in compressive imaging or fluorescence microscopy one can use random Bernoulli matrices. However, as we shall see, the use of complete randomness does not allow one to exploit the structure of the signal during the sampling procedure; it can still be taken into account after sampling (during recovery) but not as efficiently.

5.3 Taking Structure and Infinite Dimensionality into Account

The problem considered in this paper is a Type II problem, and there are several ways one can choose the sampling. However, the finite-dimensional setup in Sect. 5.1 must be extended in order to address Problem 1. The first question one may ask oneself is: how should one carry out the subsampling? Indeed, would choosing some $M \in \mathbb{N}$ and then, as suggested in Sect. 5.1, choosing uniformly at random m indices from $\{1, \ldots, M\}$ be a reasonable idea?

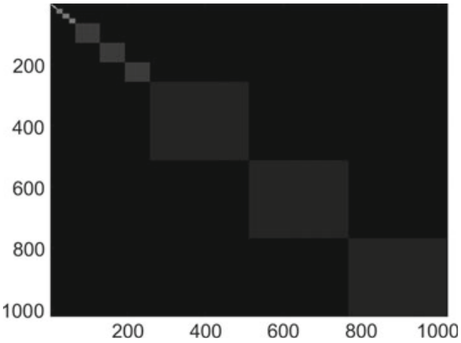


Fig. 3 Colour plot of the absolute values of the first 1024 × 1024 matrix elements of the infinite matrix U in (57) for two-dimensional Haar wavelets and Walsh functions. White colour denotes 1, whereas black illustrates 0

In order to answer this question, it may be of interest to investigate the relationship between the sampling space of Walsh functions and the reconstruction space of, for example, wavelets. Consider the infinite change of basis matrix

$$U = \begin{pmatrix} \langle \varphi_1, \zeta_1 \rangle & \langle \varphi_2, \zeta_1 \rangle & \cdots \\ \langle \varphi_1, \zeta_2 \rangle & \langle \varphi_2, \zeta_2 \rangle & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix}, \tag{57}$$

where the φ_js are the Haar wavelets and the ζ_js are the Walsh functions. In Fig. 3, the absolute values of the matrix elements (the 1024 × 1024 finite section of U) are displayed in a greyscale for the two-dimensional Walsh and Haar functions. It is evident from the figure that there is a very clear block-diagonal structure. Moreover, in Fig. 1, we can see the stable sampling rate and the absolute values of U for Daubechies 2 and Haar wavelets given sampling with Walsh functions in the one-dimensional case. It is clear that the matrix U obeys a lot of structure. Indeed, for the Walsh–Haar case, we observe perfect block diagonality, which leads to a slope of the stable sampling rate of 1. These observations can be made rigorous in the following results.

Proposition 1 *Let $\psi = \mathcal{X}_{[0,1/2]} - \mathcal{X}_{(1/2,1]}$ be the Haar wavelet. Then, we have that*

$$|\langle \psi_{R,j}, \mathrm{Wal}(n,\cdot)\rangle| = \begin{cases} 2^{-R/2} & 2^R \leq n < 2^{R+1}, 0 \leq j \leq 2^R - 1 \\ 0 & \textit{otherwise}, \end{cases} \tag{58}$$

where we recall the wavelet notation with subscripts from (27).

Note that the Haar wavelet is only defined on [0, 1] and hence does not need to be boundary corrected in contrast to higher order Daubechies wavelets.

Proposition 2 ([60]) *Let $\phi = \mathcal{X}_{[0,1]}$ be the Haar scaling function. Then, we have that the Walsh transform obeys the following block and decay structure*

$$|\langle \phi_{R,j}, \mathrm{Wal}(n,\cdot)\rangle| = \begin{cases} 2^{-R/2} & n < 2^R, 0 \leq j \leq 2^R - 1 \\ 0 & \textit{otherwise,} \end{cases} \tag{59}$$

where we recall the wavelet notation with subscripts from (27).

These results can be combined into a theorem describing the situation in two dimensions. Indeed, recall the standard two-dimensional setup for the Haar wavelet:

$$\psi_{R,j_1,j_2,l}(x_1,x_2) = \begin{cases} \phi_{R,j_1}(x_1)\psi_{R,j_2}(x_2) & l = 1 \\ \psi_{R,j_1}(x_1)\phi_{R,j_2}(x_2) & l = 2 \\ \psi_{R,j_1}(x_1)\psi_{R,j_2}(x_2) & l = 3. \end{cases} \tag{60}$$

We then get the theoretical justification for the observed structure in the change of basis matrix in Fig. 3.

Theorem 5 ([60]) *Let $\psi_{R,j_1,j_2,l}$ be the Haar wavelet defined as in* (60)*. Then, the Walsh transform has the following property. For* $0 \leq j_1, j_2 \leq 2^R - 1$*,*

$$|\langle \psi_{R,j_1,j_2,1}, \mathrm{Wal}(n_1,n_2,\cdot,\cdot)\rangle| = \begin{cases} 2^{-R} & n_1 \leq 2^R, 2^R \leq n_2 < 2^{R+1} \\ 0 & \textit{otherwise}, \end{cases} \tag{61}$$

$$|\langle \psi_{R,j_1,j_2,2}, \mathrm{Wal}(n_1,n_2,\cdot,\cdot)\rangle| = \begin{cases} 2^{-R} & 2^R \leq n_1 < 2^{R+1}, n_2 \leq 2^R \\ 0 & \textit{otherwise} \end{cases} \tag{62}$$

and for the third version

$$|\langle \psi_{R,j_1,j_2,3}, \mathrm{Wal}(n_1,n_2,\cdot,\cdot)\rangle| = \begin{cases} 2^{-R} & 2^R \leq n_1 < 2^{R+1}, 2^R \leq n < 2^{R+1} \\ 0 & \textit{otherwise.} \end{cases} \tag{63}$$

Theorem 5 describes the block-diagonal structure visualized in Fig. 3. These findings suggest that also for the compressed sensing approach it is sensible to take additional structure that can be observed for wavelets and Walsh functions into account. This motivated the introduction of an extended framework for CS [3] by generalizing the traditional CS principles of incoherence and sparsity into *asymptotic incoherence* and *asymptotic sparsity*, proposing a matched sampling procedure called *multilevel sampling*. In Sect. 5.4, we shall also discuss structured sampling in contrast with structured recovery, and implications regarding sampling operators in the context of binary measurements.

5.3.1 Multilevel Sampling

High coherence in the first few rows of U means that important information about the signal to be recovered is likely to be contained in the corresponding measurements, and thus we should fully sample these rows. Once outside this region, as coherence starts decreasing, we can subsample gradually. This is also the wisdom behind the various variable density sampling strategies, which were first introduced in [49].

Definition 3 (*Multilevel sampling*) Let $r \in \mathbb{N}$, $\mathbf{M} = (M_0, \ldots, M_r) \in \mathbb{N}^{r+1}$ with $1 \leq M_1 < \ldots < M_r$, $\mathbf{m} = (m_1, \ldots, m_r) \in \mathbb{N}^r$, with $m_k \leq M_k - M_{k-1}$, $k = 1, \ldots, r$, and that $\Omega_k \subseteq \{M_{k-1}+1, \ldots, M_k\}$, $|\Omega_k| = m_k$, are chosen uniformly at random, where $M_0 = 0$. We refer to the set $\Omega = \Omega_{\mathbf{M},\mathbf{m}} = \bigcup_{k=1}^{r} \Omega_k$ as an $(\mathbf{M}, \mathbf{m})$-multilevel sampling scheme (using r levels).

Briefly, for a vector x, the sampling amount m_k needed in each sampling band Ω_k is determined by the sparsity of x in the corresponding sparsity band Δ_k and the asymptotic coherence $\mu(P_{M_k}^{\perp} U)$.

5.3.2 Asymptotic Sparsity

Let us consider a wavelet basis indexed by one variable in the canonical way according to the different scales $\{\varphi_n\}_{n \in \mathbb{N}}$. There is a natural decomposition of $\mathbb{N}$ into finite subsets according to the wavelet scales, $\mathbb{N} = \bigcup_{k \in \mathbb{N}} \{N_{k-1}+1, \ldots, N_k\}$, where $0 = N_0 < N_1 < N_2 < \ldots$ and $\{N_{k-1}+1, \ldots, N_k\}$ is the set of indices corresponding to the kth scale. For the boundary wavelets, we have $N_i = 2^{d(J_0+i)}$. Let $x \in l^2(\mathbb{N})$ be the coefficients of a function f in this basis, $\epsilon \in (0, 1]$ and define the global sparsity, s, and the sparsity at the kth level, s_k as follows:

$$\begin{aligned} s = s(\epsilon) &= \min \left\{ n : \left\| \sum_{i \in \mathcal{N}_n} x_i \varphi_i \right\| \geq \epsilon \left\| \sum_{j=1}^{\infty} x_j \varphi_j \right\| \right\}, \\ s_k = s_k(\epsilon) &= \left| \mathcal{N}_{s(\epsilon)} \cap \{N_{k-1}+1, \ldots, N_k\} \right|, \end{aligned} \tag{64}$$

where $\mathcal{N}_n$ is the set of indices of the largest n coefficients in absolute value and $|\cdot|$ is the set cardinality. Figure 4 shows that besides being sparse, images have more structure, namely *asymptotic sparsity*, *i.e.* the relative per-level sparsity

$$s_k/(N_k - N_{k-1}) \longrightarrow 0 \tag{65}$$

rapidly as $k \to \infty$ for any fixed $\epsilon \in (0, 1]$. In particular, images are far sparser at fine scales (large k) than at coarse scales (small k). This also holds for other function systems, e.g. curvelets [15], contourlets [23] or shearlets [21]. Note that asymptotic sparsity is a rather different, and much more general structure than the connected tree structure of wavelet coefficients [53]. (64) and (65) do not assume such a tree structure, but only different local sparsities s_k at different levels.

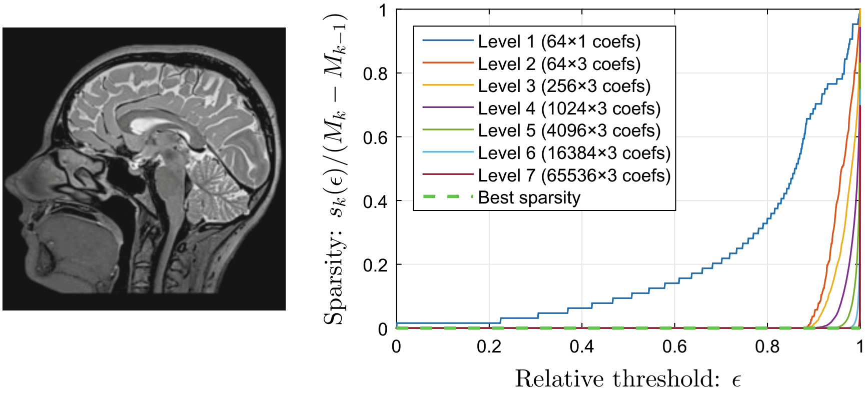


Fig. 4 Sparsity of Daubechies 4 coefficients of an MRI image, courtesy of Siemens AG. Levels correspond to wavelet scales and $s_k(\epsilon)$ is given by (64). Each curve shows the relative sparsity at level k as a function of ϵ. Their decreasing nature for increasing k confirms asymptotic sparsity (65)

Given the structure of function systems such as wavelets and their generalizations, we instead consider the notion of sparsity in levels:

Definition 4 (*Sparsity in levels*) Let x be an element of either $\mathbb{C}^N$ or $l^2(\mathbb{N})$. For $r \in \mathbb{N}$ let $\mathbf{N} = (N_0, \ldots, N_r) \in \mathbb{N}^r$ and $\mathbf{s} = (s_1, \ldots, s_r) \in \mathbb{N}^r$, with $s_k \leq N_k - N_{k-1}$, $k = 1, \ldots, r$, where $N_0 = 0$. We say that x is $(\mathbf{s}, \mathbf{N})$-sparse if, for each $k = 1, \ldots, r$ we have $|\Delta_k| \leq s_k$, where

$$\Delta_k := \operatorname{supp}(x) \cap \{N_{k-1} + 1, \ldots, N_k\}.$$

We write $\Sigma_{\mathbf{s},\mathbf{N}}$ for the set of $(\mathbf{s}, \mathbf{N})$-sparse vectors.

5.3.3 Asymptotic Incoherence

In contrast with random matrices, such as Gaussian or Bernoulli, many sampling and sparsifying operators typically found in practice yield fully coherent problems, such as the Hadamard with wavelets case discussed earlier. Indeed, Fig. 1 shows the absolute values of the entries of the matrix U with Haar and Daubechies 2 wavelets. Although there are large values of U in both cases (since U is coherent as per (54)), these are isolated to a leading submatrix. Values get asymptotically smaller once we move away from this region. This motivates the following definition.

Definition 5 (*Asymptotic incoherence*) Let $\{U_N\}$ be a sequence of isometries with $U_N \in \mathbb{C}^{N \times N}$. Then $\{U_N\}$ is asymptotically incoherent if both $\mu(P_K^{\perp} U_N)$, $\mu(U_N P_K^{\perp})$ $\to 0$ when $K \to \infty$ with $N/K = c$, for all $c \geq 1$. Conversely, if $U \in \mathcal{B}(l^2(\mathbb{N}))$, (i.e. U belongs to the space of bounded operators on $l^2(\mathbb{N})$) then we say that U is asymptotically incoherent if $\mu(P_K^{\perp} U)$, $\mu(U P_K^{\perp}) \to 0$ when $K \to \infty$.

In brief, U is asymptotically incoherent if the coherences of the matrices formed by removing either the first K rows or columns of U are small. As Fig. 1 shows the change of basis matrix U in (57) when considering Walsh functions and Haar wavelets is clearly asymptotically incoherent. However, asymptotic incoherence may not be refined enough to capture the finesses of the fine structures in the change of basis matrix U. In particular, we need the concept of local coherence, which is much more of a scalpel that allows for precise recovery guarantees.

Definition 6 (*Local coherence*) Let U be an isometry of either $\mathbb{C}^N$ or $l^2(\mathbb{N})$. If $\mathbf{M} = (M_0, \dots, M_r) \in \mathbb{N}^{r+1}$ and $\mathbf{N} = (N_0, \dots, N_r) \in \mathbb{N}^{r+1}$ with $1 \leq M_1 < \dots M_r$ and $1 \leq N_1 < \dots < N_r$ the $(k, l)^{\text{th}}$ local coherence of U with respect to $\mathbf{M}$ and $\mathbf{N}$ is given by

$$\mu_{\mathbf{M},\mathbf{N}}(k, l) = \sqrt{\mu(P_{M_k}^{M_{k-1}} U P_{N_l}^{N_{l-1}}) \cdot \mu(P_{M_k}^{M_{k-1}} U)}, \quad k, l = 1, \dots, r,$$

where $N_0 = M_0 = 0$ and P_b^a denotes the projection matrix corresponding to indices $\{a + 1, \dots, b\}$. In the case where $U \in \mathcal{B}(l^2(\mathbb{N}))$, we also define

$$\mu_{\mathbf{M},\mathbf{N}}(k, \infty) = \sqrt{\mu(P_{M_k}^{M_{k-1}} U P_{N_{r-1}}^{\perp}) \cdot \mu(P_{M_k}^{M_{k-1}} U)}, \quad k = 1, \dots, r.$$

By estimating the local coherence of U in (57) for arbitrary wavelets, we can obtain recovery guaranties for infinite-dimensional compressed sensing. These are presented in the next section.

5.3.4 Recovery Guarantees

We are stating the recovery guarantees for Walsh functions and wavelets. For this, we consider the ordering of the levels of the sampling and the reconstruction space. We get

$$\mathbf{N} = (N_0^d, N_1^d, \dots, N_r^d) = (0, 2^{d(J_0+1)}, 2^{d(J_0+2)}, \dots, 2^{d(J_0+r)}) \tag{66}$$

and

$$\begin{aligned} \mathbf{M} &= (M_0^d, M_1^d, \dots, M_{r-1}^d, M_r^d) \\ &= (0, 2^{d(J_0+1)}, 2^{d(J_0+2)}, \dots, 2^{d(J_0+r-1)}, 2^{d(J_0+r+q)}). \end{aligned} \tag{67}$$

Theorem 6 ([59]) *Let the notation be as before, i.e. let the sampling space be given by Walsh functions and the reconstruction space spanned by boundary corrected Daubechies wavelets. Additionally, let $\epsilon > 0$ and $\Omega = \Omega_{M,m}$ be a multilevel sampling scheme such that the following holds:*

1. *Let $M = M_r$, $K = \max_{k=1,\dots,r} \left\{ \frac{M_k - M_{k-1}}{m_k} \right\}$, $N = N_r$, $s = s_1 + \dots + s_r$ such that*

$$M \geq CN^2 \cdot \log_2(4NK\sqrt{s}). \tag{68}$$

2. *For each* $k = 1, \ldots, r$,

$$m_k \geq C \log(\epsilon^{-1}) \log\left(K^2 s M\right) \cdot \frac{M_k - M_{k-1}}{M_{k-1}} \cdot \left(\sum_{l=1}^{r} 2^{-d|k-l|} s_l\right). \tag{69}$$

Then with probability exceeding $1 - s\epsilon$, *any minimizer* $\xi \in \ell^1(\mathbb{N})$ *of* (56) *satisfies*

$$||\xi - x|| \leq C \cdot \left(\delta\sqrt{K}(1 + L\sqrt{s}) + \sigma_{s,N}(f)\right),$$

for some constant C, *where* $L = C \cdot \left(1 + \frac{\sqrt{\log_2(6\epsilon^{-1})}}{\log_2(4KN\sqrt{s})}\right)$. *If* $m_k = M_k - M_{k-1}$ *for* $1 \leq k \leq r$ *then this holds with probability* 1.

We see that the impact of the off block parts is exponentially decreasing. This allows us to exploit the asymptotic sparsity and reduce heavily the number of samples.

Remark 1 In Eq. (68), we see that the relation between the number of samples and coefficients is squared with an additional log factor. This quadratic term is likely to be an artefact of the proof and not sharp. In [7], it was shown that for the Fourier wavelet case this can be reduced to a linear relation, if the wavelet decays fast under the Fourier transform, i.e. if it is smooth. Unfortunately, there is no direct relation between the smoothness of the wavelet and the decay under the Walsh transform. Therefore, these results are not directly transferable and hence still open research.

5.3.5 Relation to Previous Work

It has long been known that wavelet coefficients possess additional structure beyond sparsity. In the CS context, this is the basis for structured recovery algorithms, such as model-based CS [11], Bayesian CS [42] and TurboAMP [57]. We discuss these later on. These algorithms exploit the connected tree structure of wavelet coefficients based on the 'persistence across scales' phenomenon [53]. Asymptotic sparsity assumes only asymptotic decrease of the local sparsities in the individual levels to zero. Asymptotic sparsity is more general, as the levels chosen need not correspond to the *-let (for example, wavelets or shearlets, curvelets) levels and it makes no assumption about dependencies between coefficients such as a connected tree.

A number of different characterizations of non-flat coherence patterns have been introduced in CS previously [25, 33, 61, 62]. What differentiates asymptotic incoherence is that it allows one to capture (near) block-diagonal structure inherent to *-let bases, by defining a vector of local coherence values for blocks of the coherence matrix, and specifically incorporates the asymptotic decrease of these values and the boundaries of each block. As we shall show, this is key to the practical recovery performance.

The idea of sampling the low-order coefficients of an image differently goes back to the early days of CS. In particular, Donoho considers a two-level approach for recovering wavelet coefficients in his seminal paper [24], based on acquiring the coarse scale coefficients directly. This was later extended by Tsaig & Donoho to so-called 'multiscale CS' in [63], where distinct subbands were sensed separately. See also the works by Candès and Romberg [18] and Romberg [56]. We note that the sampling schemes of [24, 56], and more recently, the 'half-half' scheme of [58] proposed for the application of CS to fluorescence microscopy are examples of two-level sampling strategies within our general framework and were analysed in detail in [3]. Our multilevel sampling extends these ideas as part of a formal framework for CS.

5.4 Points of Discussion Regarding Structure

Structured sampling and Structured Recovery. In this work, we exploit the sparsity structure at the sampling stage, by sampling asymptotically incoherent matrices, and use standard ℓ^1 minimization algorithms. Alternatively, sparsity structure can be exploited by using universal sampling matrices (e.g. random Gaussian/Bernoulli) and modified recovery algorithms which exploit the structure at the recovery stage.

Structure or Universality. The universality property of random sensing matrices (e.g. Gaussian, Bernoulli), explained later on, is a reason for their popularity in traditional CS. But is universality desirable when the signal sparsity is structured? Should one use universal matrices when there is freedom to choose the sampling operator, i.e. in Type II problems? Random matrices are largely inapplicable in Type I problems where the sampling operator yields coherent operators.

Storage and speed. Random matrices, while popular, require either large storage or are otherwise slow to generate (from a pseudorandom generator point of view), which yields slow recovery and limits the maximum signal size, which adversely affects computations. However, there exist ways to perform CS using fast transforms that emulate the usage of random matrices. Nevertheless, is addressing the speed/speed problems via fast transforms or non-random matrices sufficient?

5.4.1 Structured Sampling and Structured Recovery

The asymptotic CS framework takes into account the sparsity structure during the sampling stage via multilevel sampling of non-universal sensing matrices. Sparsity structure can also be taken into account in the recovery algorithm. A well-known example of such an approach is model-based CS [11], which assumes the signal is piecewise smooth and exploits the connected tree structure (persistence across scales) of wavelet coefficients [53] to reduce the search space of the matching pursuit algorithm [54]. The same tree structure is exploited by the class of message passing and approximate message passing algorithms [12, 27]. This can be coupled with hidden Markov trees to model the wavelet structure, such as in the Bayesian CS [42] and TurboAMP [57] algorithms. Another approach is to assign non-uniform weights to the sparsity coefficients [45], to favour the important coefficients during ℓ^1 recovery by assuming some typical decay rate of the coefficients. Another approach assumes the signal (not its representation in a sparsity basis) is sparse and random, and shows promising theoretical results when using spatially coupled matrices [26, 46, 65], yet it is unclear how a practical setup can be realized where signals are sparse in a transform domain.

The main difference is that the former approach, i.e. multilevel sampling of asymptotically incoherent matrices, incorporates sparsity structure in the sampling strategy and can use standard ℓ^1 minimization algorithms, whereas the latter approaches exploit structure by modifying the recovery algorithm and use universal sampling operators which yield uniform incoherence, e.g. random Gaussian or Bernoulli.

By using universal operators and assuming a sparsity basis, *structured recovery* is typically restricted to Type II problems, where the sensing operator can be designed (see also the remark below), and is limited by the choice of the representation system, whose structure is exploited by the modified algorithm.

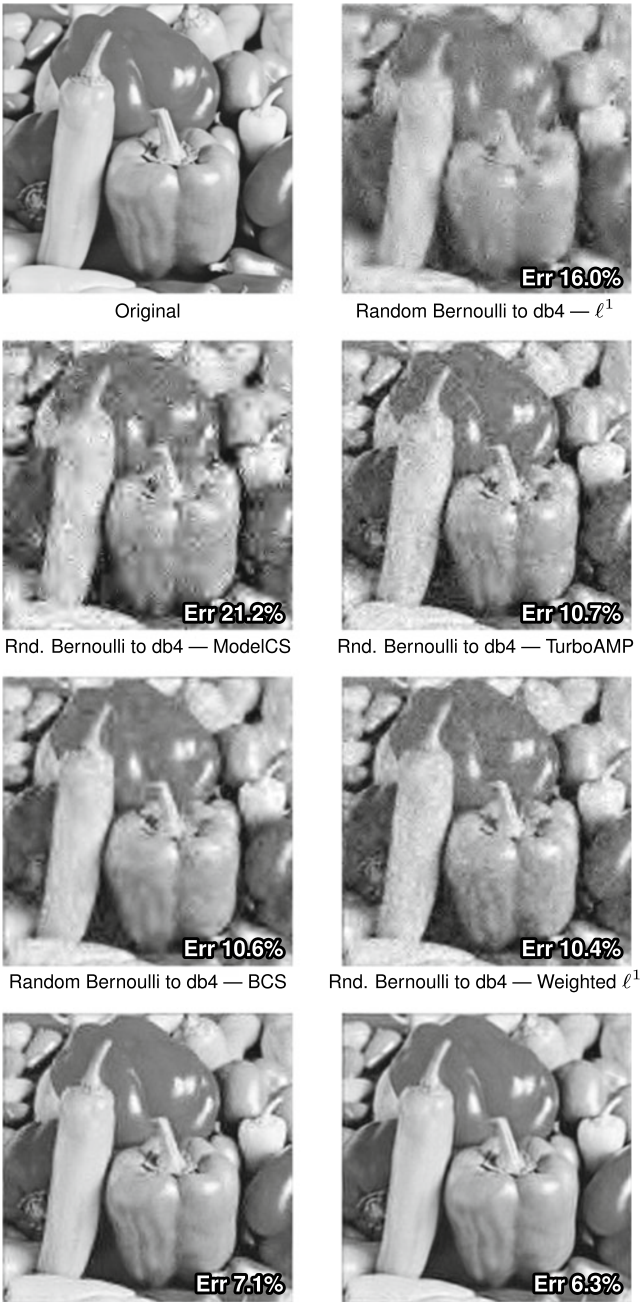
Structured sampling is flexibility with regards to the representation system and are applicable in both Type I and Type II problems.

To compare performance, we ran a set of simulations of Compressive Imaging [34, 44], which is a Type II problem, and has utilized universal sensing matrices. Binary measurements y are taken, typically using a $\{-1, 1\}^{N \times N}$ sensing matrix. Any matrix with only two values fits this setup, such as Hadamard, random Bernoulli, Sum-To-One [37], hence we can directly compare the two approaches. Figure 5 shows a representative example from our set of simulations. One can notice that asymptotic incoherence combined with multilevel sampling of highly non-universal sensing matrices (e.g. Hadamard, Fourier) allows structured sparsity to be better exploited than universal sensing matrices, even when the structure is accounted for in the recovery algorithm. The figure also shows the added benefit of being able to use a better sparsifying system, in this case curvelets.

Is it possible to combine the two approaches to leverage further gains? The structured recovery algorithms we have encountered expect the sampling operator to be incoherent with the recovery basis. Replacing those with asymptotically incoherent operators such as Hadamard or Fourier resulted in poorer performance, sometimes failing to produce a result, which isn't totally surprising given that the aforementioned structured recovery algorithms make certain assumptions about the sampling operator. Nevertheless, the successful combination of the two approaches is a promising line of investigation and is the subject of ongoing research.

5.4.2 Structure and Universality: Is Universality Desirable?

Universality is a reason for the popularity in traditional CS of random sensing matrices, e.g. Gaussian or Bernoulli. A random matrix $A \in \mathbb{C}^{m \times N}$ is universal if for every isometry $\Psi \in \mathbb{C}^{N \times N}$, the matrix $A\Psi$ satisfies the restricted isometry property [19]



Original

Random Bernoulli to db4 — ℓ^1

Rnd. Bernoulli to db4 — ModelCS

Rnd. Bernoulli to db4 — TurboAMP

Random Bernoulli to db4 — BCS

Rnd. Bernoulli to db4 — Weighted ℓ^1

Multilevel Hadamard to db4 — ℓ^1

Multilevel Had. to Curvelets — ℓ^1

Fig. 5 Compressive Imaging example. 12.5% subsampling at 256×256

with high probability. For images, a common choice is $\Psi = \Psi_{\mathrm{dwt}}^*$, the inverse wavelet transform. Universality is a key feature when the signal is sparse but possesses no further structure.

But is universality desirable in a sensing matrix when the signal is structured? First, random matrices are applicable mostly in Type II problems, where there is freedom to design the sampling operator. Hence also universal matrices are possible from a practical perspective. But should one use universal matrices there? We argue that universal matrices offer little room to exploit extra structure the signal may have, even in Type II problems.

Typical signals in practice exhibit far more structure than sparsity alone: their sparsity is asymptotic in some basis. Thus, an alternative is to use a non-universal sensing matrix, such as Hadamard, Φ_{Had}. As previously discussed and shown in Figs. 1 and 3, $U = \Phi_{\mathrm{Had}} \Psi_{\mathrm{dwt}}^*$ is completely coherent with all wavelets yet asymptotically incoherent, and thus perfectly suitable for a multilevel sampling scheme which can exploit the inherent asymptotic sparsity. This is precisely what we see in Fig. 5: a multilevel sampled Hadamard matrix can markedly outperform universal matrices in Type II problems. In Type I problems, many imposed sensing operators are non-universal and asymptotically incoherent with popular sparsity bases, and thus exploitable using multilevel sampling.

The reasons for the superior results are rooted in the incoherence structure. Universal and close to universal sensing matrices typically provide a relatively low and flat coherence pattern. This allows sparsity to be exploited by sampling uniformly at random but, by definition, these matrices cannot exploit the distinct asymptotic sparsity structure when using a typical (ℓ^1 minimization) CS reconstruction.

In contrast, when the sensing matrix provides a coherence pattern that aligns with the signal sparsity pattern, one can fruitfully exploit such structure. A multilevel sampling scheme is likely to give superior results by sampling more in the coherent regions, where the signal is also typically less sparse. The optimum sampling strategy is signal dependent. However, real-world signals, particularly images, share a fairly common structure in the wavelet domain and also in wavelet inspired representation systems. This structure allows to design variable density sampling strategies. An added benefit when this alignment exists, is that the sampling procedure allows for tailoring of the sampling pattern to target application-specific features rather than an all-round approach, e.g. recovering contours better, trading overall quality.

5.4.3 Storage and Speed: Are Non-random or Orthogonality Enough?

Random matrices require (large) storage and lack fast transforms. This limits the maximum signal resolution and yields slow recovery. For example, a 1024×1024 recovery with 25% subsampling of a random Gaussian matrix would require 2 Terabytes of free memory and $\mathcal{O}(10^{12})$ time complexity, making it impractical. The storage issue could be addressed naively, by storing only the initial seed and generating the matrix on the fly, but that makes the process orders of magnitude slower.

Both the storage and speed issues were, in fact, addressed to various extents, e.g. pseudorandom column permutations of the columns of orthogonal matrices such as (block) Hadamard or Fourier [17, 35], Kronecker products of random matrix stencils [28], or even fully orthogonal matrices such as the Sum-To-One (STOne) matrix[1] [37] which allows for a fast $\mathcal{O}(N \log N)$ transform. All these solutions in the CS context yield similar statistics to a random matrix: they become universal sampling operators.

Another solution to the storage and speed problem is to instead use structured matrices like Hadamard, DCT or DFT. These have fast transforms but also provide asymptotic incoherence with most sparsity bases, thus a multilevel subsampling scheme can be used. This yields significantly better CS recovery when compared to universal matrices, as witnessed previously, and it is also applicable to Type I problems, which impose the sensing operator.

In conclusion, the sensing matrix should contain additional structure besides simply being non-random and/or orthogonal in order to provide asymptotic incoherence. Typically, sensing and sparsifying matrices that are discrete versions of integral transforms, e.g. Fourier, wavelets etc., will provide asymptotic incoherence, but other orthogonal and structured matrices like Hadamard will do so too.

6 Conclusion

This work concerns the recovery of functions from binary measurements, that is the measurements are inner products with functions on $\{0, 1\}^N$, in the context of linear recovery using either PBDW or generalized sampling, and non-linear recovery using infinite dimensional compressed sensing. We considered the use of Walsh functions in the sampling domain and wavelets in the reconstruction domain. In the linear case, we showed that the methods rely on knowing the stable sampling rate, and we established its linearity and that it is sharp. Furthermore, we showed that generalized sampling keeps the solution in the reconstruction space which allows for improvements over PBDW in the case of highly sparse functions. In the non-linear case, we derived recovery guarantees and discussed the advantages of using Walsh functions (via the Hadamard transform) over incoherent sampling.

Acknowledgements RC acknowledges support from the Air Force Office of Scientific Research through grant FA 9550-17-1-0291. AH acknowledges support from Royal Society UK, and EPSRC UK through grant EP/L003457/1. BR acknowledges support from EPSRC UK through grants EP/N014588/1 and EP/R008272/1. LT acknowledges support from EPSRC UK through grant EP/L016516/1.

[1] The STOne matrix is an orthogonal matrix that provides universality like random matrices do. However, it was invented for many other purposes. It has a fast $\mathcal{O}(N \log N)$ transform and allows multiscale image recovery from compressive measurements: low-resolution previews can be quickly generated by applying the fast transform on the measurements directly, and high-resolution recovery is possible from the same measurements via CS solvers. In addition, it allows efficient recovery of compressive videos when sampling in a semi-random manner.

References

1. B. Adcock, A. Hansen, G. Kutyniok, J. Ma, Linear stable sampling rate: optimality of 2d wavelet reconstructions from fourier measurements. SIAM J. Math. Anal. **47**(2), 1196–1233 (2015)
2. B. Adcock, A. Hansen, C. Poon, Beyond consistent reconstructions: optimality and sharp bounds for generalized sampling, and application to the uniform resampling problem. SIAM J. Math. Anal. **45**(5), 3132–3167 (2013)
3. B. Adcock, A. Hansen, C. Poon, B. Roman, Breaking the coherence barrier: a new theory for compressed sensing. Forum Math. Sigma **5** (2017)
4. B. Adcock, A.C. Hansen, A generalized sampling theorem for stable reconstructions in arbitrary bases. J. Fourier Anal. Appl. **18**(4), 685–716 (2010)
5. B. Adcock, A.C. Hansen, Generalized sampling and infinite-dimensional compressed sensing. Found. Comput. Math. **16**(5), 1263–1323 (2016)
6. B. Adcock, A.C. Hansen, C. Poon, On optimal wavelet reconstructions from Fourier samples: linearity and universality of the stable sampling rate. Appl. Comput. Harmon. Anal. **36**(3), 387–415 (2014)
7. B. Adcock, A. C. Hansen, C. Poon, B. Roman, Breaking the coherence barrier: a new theory for compressed sensing, in *Forum of Mathematics, Sigma*, volume 5. Cambridge University Press (2017)
8. A. Aldroubi, M. Unser, A general sampling theory for nonideal acquisition devices. IEEE Trans. Signal Process. **42**(11), 2915–2925 (1994)
9. V. Antun, Coherence estimates between hadamard matrices and daubechies wavelets. Master's thesis, University of Oslo (2016)
10. M. Bachmayr, A. Cohen, R. DeVore, G. Migliorati, Sparse polynomial approximation of parametric elliptic pdes. part ii: lognormal coefficients. ESAIM: Math. Modell. Numer. Anal. **51**(1), 341–363 (2017)
11. R. Baraniuk, V. Cevher, M. Duarte, C. Hedge, Model-based compressive sensing. IEEE T Inf. Th. **56**(4) (2010)
12. D. Baron, S. Sarvotham, R. Baraniuk, Bayesian compressive sensing via belief propagation. IEEE T Sig. Proc. **58**(1) (2010)
13. P. Binev, A. Cohen, W. Dahmen, R. DeVore, G. Petrova, P. Wojtaszczyk, Data assimilation in reduced modeling. SIAM/ASA J. Uncertain. Quantif. **5**(1), 1–29 (2017)
14. A. Böttcher, Infinite matrices and projection methods: in lectures on operator theory and its applications, fields inst. monogr. Amer. Math. Soc. (3), 1–72 (1996)
15. E. Candès, D. Donoho, Recovering edges in ill-posed inverse problems: optimality of curvelet frames. Ann. Statist. **30**(3) (2002)
16. E. Candès, Y. Plan, A probabilistic and RIPless theory of compressed sensing. IEEE T Inf. Th. **57**(11) (2011)
17. E. Candès, J. Romberg, Robust signal recovery from incomplete observations, in *IEEE International Conference on Image Processing* (2006)
18. E. Candès, J. Romberg, Sparsity and incoherence in compressive sampling. Inverse Problems **23**(3) (2007)
19. E. Candès, J. Romberg, T. Tao, Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. IEEE T Inf. Th. **52**(2) (2006)
20. A. Cohen, I. Daubechies, P. Vial, Wavelets on the interval and fast wavelet transforms. Comput. Harmon. Anal. **1**(1), 54–81 (1993)
21. S. Dahlke, G. Kutyniok, P. Maass, C. Sagiv, H.-G. Stark, G. Teschke, The uncertainty principle associated with the continuous shearlet transform. Int. J. Wavelets Multiresolut. Inf. Process. **6**(2) (2008)
22. R. DeVore, G. Petrova, P. Wojtaszczyk, Data assimilation and sampling in banach spaces. Calcolo **54**(3), 963–1007 (2017)
23. M. Do, M. Vetterli, The contourlet transform: an efficient directional multiresolution image representation. IEEE T Image Proc. **14**(12) (2005)

24. D. Donoho, Compressed sensing. IEEE T Inf. Th. **52**(4) (2006)
25. D. Donoho, M. Elad, Optimally sparse representation in general (non-orthogonal) dictionaries via ℓ_1 minimization. Proc. Natl. Acad. Sci. USA **100** (2003)
26. D. Donoho, A. Javanmard, A. Montanari, Information-theoretically optimal compressed sensing via spatial coupling and approximate message passing. IEEE T Inf. Th. **59**(11) (2013)
27. D. Donoho, A. Maleki, A. Montanari, Message-passing algorithms for compressed sensing. Proc. Natl Acad. Sci. USA **106**(45) (2009)
28. M. Duarte, R. Baraniuk, Kronecker compressive sensing. IEEE T Image Proc. **21**(2) (2012)
29. T. Dvorkind, Y.C. Eldar, Robust and consistent sampling. IEEE Signal Process. Lett. **16**(9), 739–742 (2009)
30. Y.C. Eldar, Sampling with arbitrary sampling and reconstruction spaces and oblique dual frame vectors. J. Fourier Anal. Appl. **9**(1), 77–96 (2003)
31. Y.C. Eldar, *Sampling without Input Constraints: Consistent Reconstruction in Arbitrary Spaces* (Sampling, Wavelets and Tomography, 2003)
32. Y.C. Eldar, T. Werther, General framework for consistent sampling in hilbert spaces. Int. J. Wavelets Multiresolut. Inf. Process. **3**(4), 497–509 (2005)
33. S. Foucart , H. Rauhut. *A Mathematical Introduction to Compressive Sensing*. Birkhäuser (2013)
34. S. Foucart, H. Rauhut, *A Mathematical Introduction to Compressive Sensing* (Springer Science+Business Media, New York, 2013)
35. L. Gan, T. Do, and T. Tran, Fast compressive imaging using scrambled hadamard ensemble. Proc. Eur. Signal Proc. Conf. (2008)
36. E. Gauss, *Walsh Funktionen für Ingenieure und Naturwissenschaftler* (Springer Fachmedien, Wiesbaden, 1994)
37. T. Goldstein, L. Xu, K. Kelly, R. Baraniuk, The stone transform: multi-resolution image enhancement and real-time compressive video. arXiv:1311.3405 (2013)
38. K. Gröchenig, Z. Rzeszotnik, T. Strohmer, Quantitative estimates for the finite section method and banach algebras of matrices. Integr. Equ. Oper. Theory **2**(67), 183–202 (2011)
39. A. Hansen, On the approximation of spectra of linear operators on hilbert spaces. J. Funct. Anal. **8**(254), 2092–2126 (2008)
40. A.C. Hansen, On the solvability complexity index, the n-pseudospectrum and approximations of spectra of operators. J. Amer. Math. Soc. **24**(1), 81–124 (2011)
41. A. C. Hansen, L. Thesing, On the stable sampling rate for binary measurements and wavelet reconstruction. preprint (2017)
42. L. He, L. Carin, Exploiting structure in wavelet-based Bayesian compressive sensing. IEEE T Sig. Proc. **57**(9) (2009)
43. T. Hrycak, K. Gröchenig, Pseudospectral fourier reconstruction with the modified inverse polynomial reconstruction method. J. Comput. Phys. **229**(3), 933–946 (2010)
44. G. Huang, H. Jiang, K. Matthews, P. Wilford, Lensless imaging by compressive sensing. IEEE Intl. Conf. Image Proc. (2013)
45. M. Khajehnejad, W. Xu, A. Avestimehr, B. Hassibi, Analyzing weighted ℓ_1 minimization for sparse recovery with nonuniform sparse models. IEEE T Sig Proc. **59**(5) (May 2011)
46. F. Krzakala, M. Mézard, F. Sausset, Y. Sun, L. Zdeborová, Statistical-physics-based reconstruction in compressed sensing. Phys. Rev. X **2** (May 2012)
47. G. Kutyniok, W.-Q. Lim, Optimal compressive imaging of fourier data. SIAM J. Imaging Sci. **11**(1), 507–546 (2018)
48. M. Lindner, *Infinite Matrices and their Finite Sections: An Introduction to the Limit Operator Method* (Birkhäuser Verlag, Basel, 2006)
49. M. Lustig, D. Donoho, J. Pauly, Sparse MRI: the application of compressed sensing for rapid MRI imaging. Magn. Reson. Imaging **58**(6) (2007)
50. J. Ma, Generalized sampling reconstruction from fourier measurements using compactly supported shearlets. Appl. Comput. Harmon. Anal. (2015)
51. Y. Maday, A.T. Patera, J.D. Penn, M. Yano, A parameterized-background data-weak approach to variational data assimilation: formulation, analysis, and application to acoustics. Int. J. Numer. Methods Eng. **102**(5), 933–965 (2015)

52. S. Mallat, *A Wavelet Tour of Signal Processing* (Academic Press, San Diego, 1998)
53. S. Mallat, *A Wavelet Tour of Signal Processing: The Sparse Way*, 3nd edn. (Academic Press, 2009)
54. D. Needell, J. Tropp, CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. Appl. Comput. Harmonic Anal. **26**(3) (2009)
55. C. Poon, Structure dependent sampling in compressed sensing: theoretical guarantees for tight frames. Appl. Comput. Harmonic Anal. **42**(3), 402–451 (2017)
56. J. Romberg, Imaging via compressive sampling. IEEE Sig. Proc. Mag. **25**(2) (2008)
57. S. Som, P. Schniter, Compressive imaging using approximate message passing and a markov-tree prior. IEEE T Sig. Proc. **60**(7) (2012)
58. V. Studer, J. Bobin, M. Chahid, H.S. Mousavi, E. Candes, M. Dahan, Compressive fluorescence microscopy for biological and hyperspectral imaging. Proc. Natl Acad. Sci. **109**(26), E1679–E1687 (2012)
59. L. Thesing, A. Hansen. Non uniform recovery guarantees for binary measurements and wavelet reconstruction. To appear
60. L. Thesing, A. Hansen, Linear reconstructions and the analysis of the stable sampling rate. Sampl. Theory Image Process. **17**(1), 103–126 (2018)
61. J. Tropp, Greed is good: algorithmic results for sparse approximation. IEEE T Inf. Th. **50**(10) (2004)
62. J. Tropp, Just relax: convex programming methods for identifying sparse signals in noise. IEEE T Inf. Th. **52**(3) (2006)
63. Y. Tsaig, D. Donoho, Extensions of compressed sensing. Signal Process **86**, 3 (2006)
64. M. Unser, J. Zerubia, A generalized sampling theory without band-limiting constraints. IEEE Trans. Circuits Syst. II. **45**(8), 959–969 (1998)
65. Y. Wu, S. Verdu, Rényi information dimension: fundamental limits of almost lossless analog compression. IEEE T Inf. Th. **56**(8) (2010)

Classification Scheme for Binary Data with Extensions

Denali Molitor, Deanna Needell, Aaron Nelson, Rayan Saab and Palina Salanevich

Abstract In this chapter, we present a simple classification scheme that utilizes only 1-bit measurements of the training and testing data. Our method is intended to be efficient in terms of computation and storage while also allowing for a rigorous mathematical analysis. After providing some motivation, we present our method and analyze its performance for a simple data model. We also discuss extensions of the method to the hierarchical data setting, and include some further implementation considerations. Experimental evidence provided in this chapter demonstrates that our methods yield accurate classification on a variety of synthetic and real data.

1 Introduction

In this work, we discuss the problem of classification. More precisely, a supervised learning problem where one is given labeled *training data*, and from that data one wishes to determine a rule with which to accurately assign labels to unlabeled future *test data* points is considered. We focus on the setting where either by design or by

The views expressed in this article are those of the authors and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the U.S. Government.

D. Molitor · D. Needell · P. Salanevich
University of California, Los Angeles, CA, USA
e-mail: dmolitor@math.ucla.edu

D. Needell
e-mail: deanna@math.ucla.edu

P. Salanevich
e-mail: psalanevich@math.ucla.edu

A. Nelson (✉) · R. Saab
University of California, San Diego, CA, USA
e-mail: aan031@ucsd.edu

R. Saab
e-mail: rsaab@ucsd.edu

H. Boche et al. (eds.), *Compressed Sensing and Its Applications*,
Applied and Numerical Harmonic Analysis,
https://doi.org/10.1007/978-3-319-73074-5_4

application, the data is only available in a *binary* representation. Such representations may be obtained through compressive sampling, as in applications that have restricted bandwidth or energy constraints [19], or may be utilized to take advantage of simpler, faster, and cheaper hardware implementations [31, 37]. In general, compression and coarse quantization can be appealing due to efficient storage and computation. Additionally, such representations can still be utilized when performing inference tasks, e.g., see preliminary work of [4, 22, 26, 27]. This chapter presents a framework for learning inferences from highly quantized (single bit) data representations, with the key example being classification. Let us begin with some mathematical tools and notation.

1.1 Notation and Setup

Let $\{x_i\}_{i=1}^p \subset \mathbb{R}^n$ be a data set represented in matrix form

$$X = [x_1\ x_2\ \dots\ x_p] \in \mathbb{R}^{n\times p}.$$

Let $A : \mathbb{R}^n \to \mathbb{R}^m$ be a linear map, and denote by sign : $\mathbb{R} \to \mathbb{R}$ the sign operator defined by

$$\mathrm{sign}(a) := \begin{cases} 1 & a \geq 0 \\ -1 & a < 0. \end{cases}$$

We generalize this operator for matrices elementwise, where for an m by p matrix M, and $(i, j) \in [m] \times [p]$, we define $\mathrm{sign}(M)$ as the $m \times p$ matrix with entries

$$(\mathrm{sign}(M))_{i,j} := \mathrm{sign}(M_{i,j}).$$

We now consider the setting where our method has access to training data of the form $Q = \mathrm{sign}(AX)$, along with the labels $b = (b_1,\ \dots\ , b_p) \in \{1, \dots , G\}^p$, that identify each point x_i as belonging to one of the G possible classes. The rows of the $m \times n$ matrix A correspond to m *hyperplanes* in $\mathbb{R}^n$ and the sign information in Q captures on which side of the hyperplane each data point lies.

Throughout this chapter, A is assumed to have independent identically distributed standard Gaussian entries. We will present an approach from [41] that, given Q and b, allows for classification of a new unlabeled data point $x \in \mathbb{R}^n$ from its binary measurements $\mathrm{sign}(Ax)$, and we will discuss various extensions and open problems associated with it.

1.2 Related Work and Background

We briefly mention here a few related topics that motivated the method proposed in [41]. We encourage the reader to see included references and others therein for more thorough background and details of these large areas of work.

Support vector machines (SVM) [3, 14, 24, 32, 47] are a popular method for classification. From labeled training data, SVMs seek an optimal hyperplane (or multiple hyperplanes) that separates the data or maximizes the geometric margin between the classes in the case where the data is not linearly separable. The approach described in this chapter is similar in flavor, but instead of optimizing hyperplane parameters to fit the data, it uses many random hyperplanes to identify separation of the data, and aggregates that information to decide upon a label.

The use of dimension reduction, that is the transformation of high dimensional data into geometrically similar low dimensional representations, appears in numerous contexts. The Johnson–Lindenstrauss Lemma guarantees the existence of a map that embeds p points into $O(\epsilon^{-2}\log(p))$ dimensions while approximately (up to ϵ) preserving the geometry [30]. In fact, a Johnson–Lindenstrauss map can be linear and can be obtained (with high probability) via a random draw from an appropriate distribution. Such random linear maps include those associated with Gaussian or subgaussian matrices and those resulting from selecting random rows of the discrete Fourier transform [1, 2, 5, 15, 35, 46]. Constructions of such maps play a crucial role in the field of *compressed sensing*, where they are used to sample high dimensional signals, yielding effective sampling rates that break the traditional Nyquist bounds [12, 13, 16]. Mathematically, for a signal $x \in \mathbb{R}^n$, one uses a measurement matrix $A \in \mathbb{R}^{m\times n}$ to acquire (possibly noisy) measurements of the form $y = Ax + z$, and the goal is to recover the signal x. For Johnson–Lindenstrauss type matrices A, the assumption that turns this ill-posed highly underdetermined problem into a well-posed problem is that the signal x is s-sparse, meaning that $\|x\|_0 := |\operatorname{supp}(x)| = s \ll n$.

For any digital compression scheme to be practical, one must consider the need to *quantize*, that is, to restrict data to a discrete set of values. Pushing quantization to the extreme in compressed sensing, the so-called *1-bit compressed sensing* problem captures only a single bit per measurement and asks to recover the measured signal x [4]. Formulated mathematically, one acquires measurements of the form $y = \operatorname{sign}(Ax)$, possibly with pre- or post-quantization noise. Clearly the normalization of x is lost under such scalar-invariant measurements, but under a norm assumption, efficient methods have been developed that accurately recover x [21, 29, 31, 42, 43, 55]. This normalization assumption can be overcome by introducing *dithers* to the measurements and modifying the methods [6, 34]. These branches of work have sparked recent interest in *binary embeddings*, those that map vectors to the binary cube while preserving angular information among the mapped vectors [7, 17, 20, 44, 53, 54]. The 1-bit compressed sensing problem and these binary embeddings motivate the work presented in this chapter, although the end goal of our consideration is classification rather than reconstruction.

Lastly, no modern chapter on classification would be complete without mentioning the burgeoning work on *deep learning*, which learns data representations using multiple levels of abstraction, usually referred to as layers or levels. Each layer can be viewed as a function that learns its parameters from the training data, so that its input data is transformed into a slightly more abstract and composite representation. From the composition of these functions, a (neural) network is constructed that solves the desired learning task (e.g., classification) by extracting relevant features of data. With the abundance of large data sets, these neural networks have become state of the art, yielding often astoundingly good results and techniques that continue to improve [33, 45, 50, 51]. On the other hand, although there is theoretical analysis (see e.g., [38, 40]), their success is often difficult to quantitatively analyze and interpret [56]. While the work presented in this chapter may be similar in flavor at a high scale, the aim is quite different—we intend to develop a simple approach to classification that allows for quantitative success bounds and simple geometric interpretability.

1.3 Organization

The remainder of the chapter is organized as follows. Section 2 motivates and describes the classification method. Section 2.1 provides an analysis for a simple data model, bounding the probability that a new test point is correctly labeled. Section 2.2 presents experimental results for the method on several synthetic and real examples. Section 3 extends the method to the setting of hierarchical classification, where the class labels have additional structure. Section 4 proposes several implementation variants that help guide parameter selection. We conclude with some final remarks in Sect. 5.

2 Simple Classification Approach

We next turn to a description of the classification method put forth in [41]. Let us first build some intuition for the approach. Consider the two-dimensional data X shown in the left plot of Fig. 1, consisting of three labeled classes (green, blue, red), and suppose we only have access to the binary data $Q = \text{sign}(AX)$. Note that Q contains information giving the side of each hyperplane (corresponding to the rows of A) on which each data point lies. Consider the four hyperplanes shown in the same plot, and suppose that we are given the new test point x (which visually appears to belong to the blue class) and its binary data $q = \text{sign}(Ax)$. Then, at first glance, a reasonable algorithm is to simply cycle through the hyperplanes and decide which class x matches most often. For example, for the hyperplane colored purple in the plot, x has the same sign (i.e., lies on the same side) as the blue and green classes.

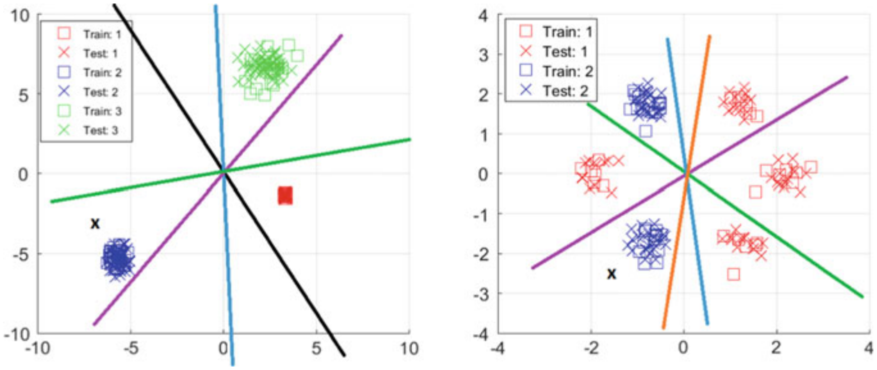


Fig. 1 Two motivating examples for the classification method

For the black hyperplane, x only matches the blue class, and so on. For this example, x will clearly match the blue class most often, and we could correctly assign it that label.

However, this may not work well for more complicated data. As an example, consider data as shown in the right plot of Fig. 1. Assuming that each point cloud has (approximately) the same cardinality, we have that for the blue and red hyperplanes, $\frac{2}{3}$ of the points lying in the same half-space as x are from the blue class. At the same time, for the purple and green hyperplanes $\frac{2}{3}$ of the points lying in the same half-space as x are from the red class. Thus, it is not possible to correctly classify x using just the information provided by individual hyperplanes. If we now consider hyperplane *pairs*, and find the class label that x most often agrees with (that is, we we find the class with points that most often share cones bounded by the pairs of hyperplanes with x), we are able to correctly classify x. Indeed, we have the following:

Pair	Blue class in the cone	Red class in the cone
Blue and red	$\frac{2}{3}$	$\frac{1}{3}$
Blue and purple	1	0
Blue and green	$\frac{1}{2}$	$\frac{1}{2}$
Red and purple	1	0
Red and green	$\frac{1}{2}$	$\frac{1}{2}$
Purple and green	$\frac{1}{2}$	$\frac{1}{2}$
Overall	$\frac{25}{6}$	$\frac{11}{6}$

We now describe the approach more formally. Again, denote by $X \in \mathbb{R}^{n \times p}$ the matrix whose columns contain the data points. Let $A \in \mathbb{R}^{m \times n}$ have rows corresponding to the normal vectors of m randomly oriented hyperplanes that pass through the origin (e.g., A could have i.i.d. Gaussian entries), and $Q = \text{sign}(AX)$ denote the binary sign information. Then, the training algorithm proceeds in L "levels". In the

ℓth level, m index sets $\Lambda_{\ell,i} \subset [m]$, $|\Lambda_{\ell,i}| = \ell$, $i = 1, \ldots, m$, are randomly selected, and each index set corresponds to an ℓ-tuple of hyperplanes. Let $Q^{\Lambda_{\ell,i}} \in \mathbb{R}^{\ell \times p}$ be the submatrix consisting of the rows of Q whose indices belong to $\Lambda_{\ell,i}$. Each column of $Q^{\Lambda_{\ell,i}}$ then gives a sign pattern of length ℓ corresponding to a training data point and the ℓ hyperplanes contained in $\Lambda_{\ell,i}$. Let us denote the number of different sign patterns of training points corresponding to $\Lambda_{\ell,i}$ (that is, number of different columns of $Q^{\Lambda_{\ell,i}}$) by $T_{\ell i}$.

At a given level ℓ, for the tth sign pattern and gth class, a *membership index* parameter $r(\ell, i, t, g)$ that uses knowledge of the number of training points in class g having the tth sign pattern, is calculated for every ℓ-tuple $\Lambda_{\ell,i}$. Below, $P_{g|t}= P_{g|t}(\Lambda_{\ell,i})$ denotes the number of training points from the gth class with the tth sign pattern at the ith set selection in the ℓth level:

$$r(\ell, i, t, g) = \frac{P_{g|t}}{\sum_{j=1}^{G} P_{j|t}} \frac{\sum_{j=1}^{G} |P_{g|t} - P_{j|t}|}{\sum_{j=1}^{G} P_{j|t}}. \tag{1}$$

Note that the first fraction in (1) indicates the proportion of training points in class g out of all points with sign pattern t (at the ℓth level and ith set selection). The second fraction in (1) is a balancing term that gives more weight to group g when that group is much different in size than the others with the same sign pattern. Intuitively, larger values of $r(\ell, i, t, g)$ suggest that the tth sign pattern is more heavily dominated by class g; thus, if a signal with unknown label corresponds to the tth sign pattern, we will be more likely to classify it into the gth class. With this intuition, we can then assign a label to a new test point x using its binary data $q = \text{sign}(Ax)$. For each class g, we simply sum the membership index function values over all ℓ, i, and t, for those sign patterns t that match the sign pattern of the new test point x (which is known via the data q). Thus, we obtain a value for each class g and the label for x is then decided by simply taking the class g corresponding to the largest sum. The training and classification portions of this method are summarized in Algorithms 1 and 2.

Algorithm 1: Training

```
Input: binary training data $Q$, training labels $b$, number of classes $G$, number of layers $L$
for $\ell$ from 1 to L, i from 1 to m do
    select: Randomly select $\Lambda_{\ell,i} \subset [m]$, $|\Lambda_{\ell,i}| = \ell$
    determine: Determine the $T_{\ell,i} \in \mathbb{N}$ unique column sign patterns in $Q^{\Lambda_{\ell,i}}$
    for t from 1 to $T_{\ell,i}$, g from 1 to G do
        compute: Compute $r(\ell, i, t, g)$ by (1)
    end
end
```

Algorithm 2: Classification

```
Input: binary data $q$, number of classes $G$, number of layers $L$, learned parameters
        $r(\ell, i, t, g)$, $T_{\ell,i}$, and $\Lambda_{\ell,i}$ from Algorithm 1
Initialize: $\tilde{r}(g) = 0$ for $g = 1, \ldots, G$.
for $\ell$ from 1 to L, i from 1 to m do
    identify: Identify the pattern $t^\star \in [T_{\ell,i}]$ to which $q^{\Lambda_{\ell,i}}$ corresponds
    for g from 1 to G do
        update: $\tilde{r}(g) = \tilde{r}(g) + r(\ell, i, t^\star, g)$
    end
end
scale: Set $\tilde{r}(g) = \frac{\tilde{r}(g)}{Lm}$ for $g = 1, \ldots, G$
classify: $\widehat{b_x} = \text{argmax}_{g\in\{1,\ldots,G\}} \{\tilde{r}(g)\}$
```

2.1 Analytical Justification

One of the benefits of this simple approach to classification is that it can be mathematically analyzed and understood. Indeed, we present here a result from [41] that bounds the probability of accurate classification for a simple data model, showcasing the potential of this method to be rigorously supported mathematically. Here, we focus on the setting where the signals are two-dimensional, belonging to one of two classes, and consider a single level (i.e., $L = 1$, $n = 2$, and $G = 2$). For simplicity of analysis, we consider the continuous setting and assume the true classes G_1 and G_2 are two disjoint *cones* in $\mathbb{R}^2$ in which the training data lies in a uniform density. See Fig. 2 for a visualization of the setup; we will describe the relevant parameters next.

Let A_1 denote the angular measure of G_1 and define A_2 similarly for G_2. Also, define A_{12} as the angle between classes G_1 and G_2. Assume all angles are such that no hyperplane will intersect both classes at once, i.e., $A_{12} + A_1 + A_2 \leq \pi$. Suppose

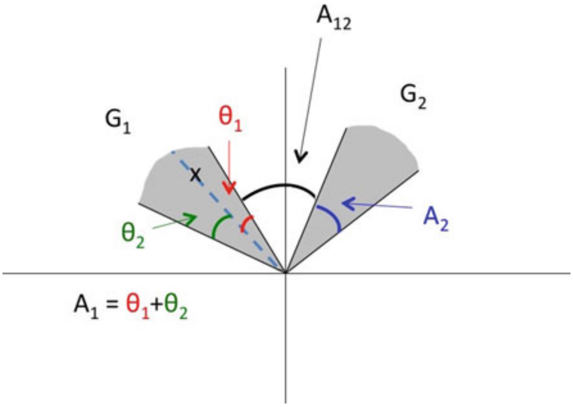


Fig. 2 Visualization of the analysis setup for two classes in two dimensions

that the test point $x \in G_1$, partitioning A_1 into two disjoint pieces, yielding angles θ_1 and θ_2, where $A_1 = \theta_1 + \theta_2$ (see Fig. 2).

The membership index parameter (1) is still used; however, in this continuous setting, we use the analogous formula with angles instead of numbers of training points:

$$r(\ell, i, t, g) = \frac{A_{g|t}}{\sum_{j=1}^{G} A_{j|t}} \frac{\sum_{j=1}^{G} |A_{g|t} - A_{j|t}|}{\sum_{j=1}^{G} A_{j|t}}, \tag{2}$$

where $A_{g|t}$ denotes the angle of class g with the tth sign pattern for the ith ℓ-tuple of hyperplanes in the ℓth layer. Denote by $t_i^\star$ the sign pattern of the test point x with the ith hyperplane at the first level (i.e., $\ell = 1$). Let $\widehat{b_x}$ denote the classification label assigned to x by Algorithm 2. Then Theorem 1 below describes the probability that x is classified correctly with $\widehat{b_x} = 1$. For simplicity, Theorem 1 is stated under the assumption that $A_1 = A_2$ and the test point x lies in the middle of class G_1 (i.e., $\theta_1 = \theta_2$). The analysis follows similarly for the general case, with more tedious computations and messier results, see [41] for the proof details.

Theorem 1 (From [41]) *Let the classes G_1 and G_2 be two cones in $\mathbb{R}^2$ defined by angular measures A_1 and A_2, respectively, and suppose regions of the same angular measure have the same density of training points. Suppose $A_1 = A_2$, $\theta_1 = \theta_2$, and $A_{12} + A_1 + A_2 \leq \pi$. Then, the probability that a data point $x \in G_1$ gets classified in class G_1 by Algorithms 1 and 2 using a single level and a measurement matrix $A \in \mathbb{R}^{m\times 2}$ with independent standard Gaussian entries is bounded as follows,*

$$\mathbb{P}[\widehat{b_x} = 1] \geq 1 - \sum_{\substack{j=0 \\ j+k_{1,\theta_1}+k_{1,\theta_2}+k_2+k=m,\ k_{1,\theta_2}\geq 9(j+k_{1,\theta_1})}}^{m} \sum_{k_{1,\theta_1}=0}^{m} \sum_{k_{1,\theta_2}=0}^{m} \sum_{k_2=0}^{m} \sum_{k=0}^{m} \binom{m}{j, k_{1,\theta_1}, k_{1,\theta_2}, k_2, k}$$
$$\times \left(\frac{A_{12}}{\pi}\right)^{j} \left(\frac{A_1}{2\pi}\right)^{k_{1,\theta_1}+k_{1,\theta_2}} \left(\frac{A_1}{\pi}\right)^{k_2} \left(\frac{\pi - 2A_1 - A_{12}}{\pi}\right)^{k}. \tag{3}$$

Although the bound on the probability given in this theorem is quite cumbersome, some useful properties are immediate. For example, this probability bound tends to 1 as m grows large. Indeed, the following two corollaries show precisely this behavior.

Corollary 1 *Consider the setup of Theorem 1. Suppose $A_{12} \geq A_1$ and $A_{12} \geq \pi - 2A_1 - A_{12}$. Then $\mathbb{P}[\widehat{b_x} = 1] \to 1$ as $m \to \infty$.*

Corollary 2 *Consider the setup of Theorem 1. Suppose $A_1 + A_{12} > 0.58\pi$ and $A_{12} + \frac{3}{4}A_1 \leq \frac{\pi}{2}$. Then $\mathbb{P}[\widehat{b_x} = 1] \to 1$ as $m \to \infty$.*

These asymptotic results are noteworthy, but of course one more importantly would like to know at what rate this probability increases to 1 as a function of the

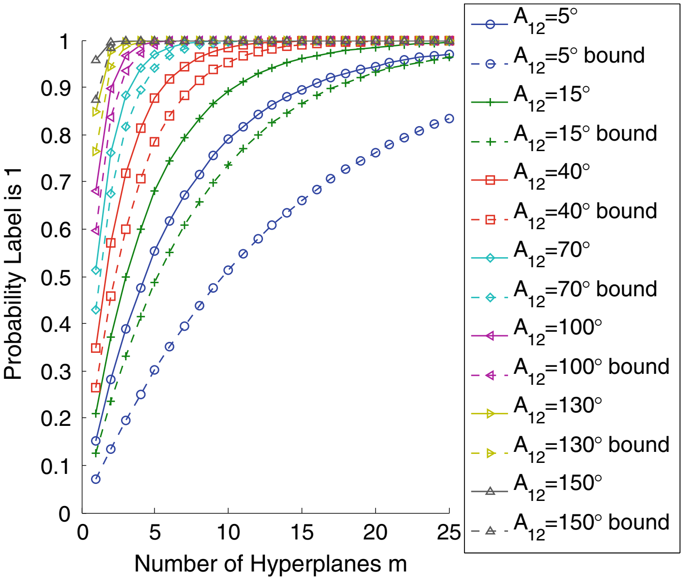


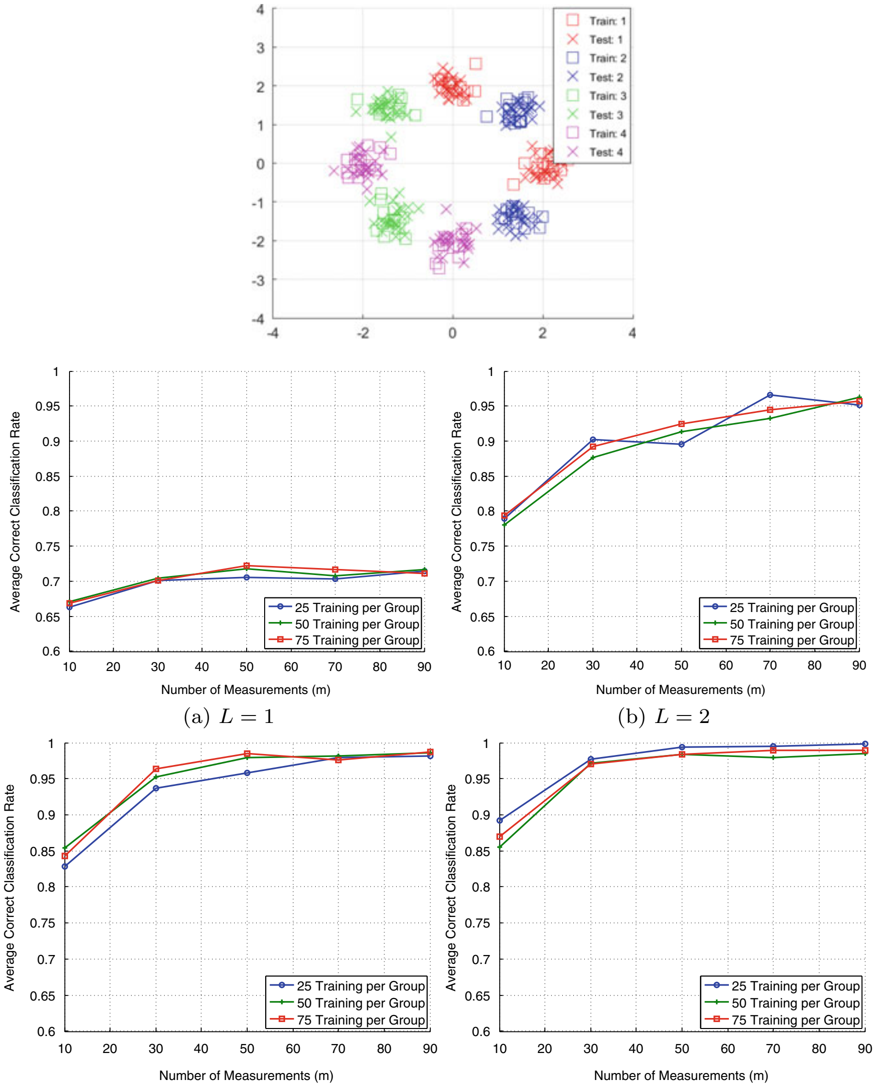
Fig. 3 $\mathbb{P}[\widehat{b_x} = 1]$ versus the number of hyperplanes m when A_{12} is varied (see legend), $A_1 = A_2 = 15°$, and $\theta_1 = \theta_2 = 7.5°$. The solid lines indicate the true (simulated) probability and the dashed lines indicate the bound (3) provided in Theorem 1

number m of hyperplanes. Indeed, it can be seen from the proofs in [41] that the probability converges to 1 exponentially in m. To illustrate this, the rates of the bound provided by Theorem 1 are displayed in Fig. 3 along with the (simulated) true value of $\mathbb{P}[\widehat{b_x} = 1]$. Although the bound is clearly not sharp, it exhibits the same overall behavior as the true probability of accurate classification.

2.2 Experimental Results

We present here a small collection of experimental results for the classification method that show its performance on synthetic and real data. The first experiment considers synthetic data consisting of eight Gaussian clouds, belonging to four classes. A new test point is drawn according to one of these distributions and is then classified by the method. The average correct classification rate (where the "correct" label is deemed to be the label matching the point cloud from which the test point x was drawn) is calculated over 50 trials and displayed. Figure 4 showcases the classification accuracy for various numbers of levels L, showing that as one expects, more levels are needed for accurate classification for complicated data geometries.

Next, we test the method on several real data sets. First, Fig. 5 shows average accuracy results for classifying the "0" versus "1" handwritten digits from the MNIST



(a) $L = 1$

(b) $L = 2$

(c) $L = 3$

(d) $L = 4$

Fig. 4 Data (top) is eight Gaussian clouds and four classes ($G = 4$), $L = 1, \ldots, 4$, $n = 2$, 50 test points per group, and 30 trials of randomly generating A. Average correct classification rate versus m and for the indicated number of training points per class for: (middle left) $L = 1$, (middle right) $L = 2$, (bottom left) $L = 3$, (bottom right) $L = 4$

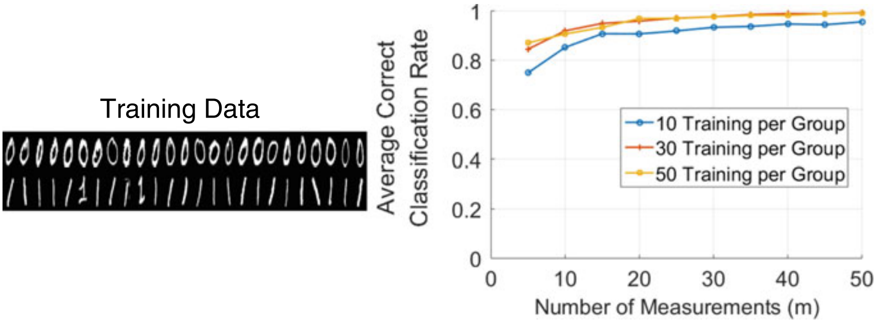


Fig. 5 Classification experiment using the handwritten "0" and "1" digit images from the MNIST data set, with 50 test points per group, $L = 1$, $n = 28 \times 28 = 784$, and 30 trials of randomly generating A. Left: example data. Right: average correct classification rate versus m and for the indicated number of training points per class

data set [36]. For this data, we only needed one level to get accurate results, perhaps because the images of the "0" and "1" digits are well separated in space. Not surprisingly, when classifying all ten digits, more levels are needed in order to obtain decent accuracy; see Fig. 6.

We also tested the method on the problem of facial classification, using the YaleB data set [8–10, 28]. Figure 7 shows classification results using six layers. Note that the results appear noisier due to the smaller size of the data set.

Lastly, we tested the method on recently acquired survey data from patients with Lyme disease, from the MyLymeData project hosted by lymedisease.org that now has

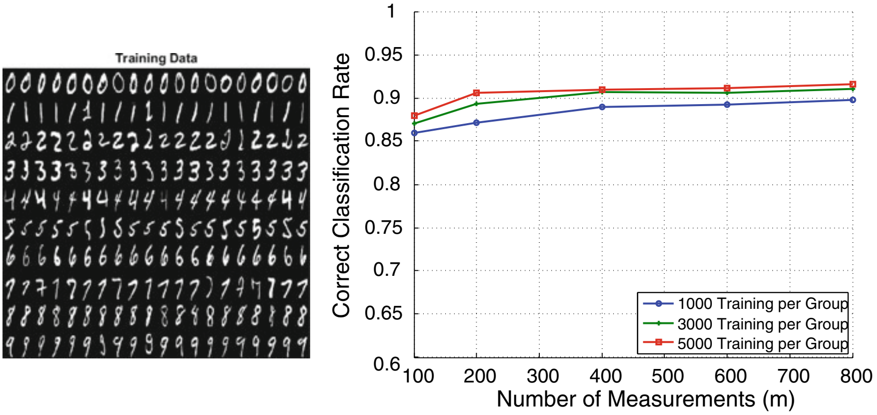


Fig. 6 Correct classification rate (right) versus m when using all ten (0–9) handwritten digits from the MNIST data set (left) with 1,000, 3,000, and 5,000 training points per group, $L = 18$, $n = 28 \times 28 = 784$, 800 test points per group (8,000 total), and a single instance of randomly generating A

over 10,000 patients enrolled. Figure 8 shows classification results using the survey responses for the symptom-related questions as our data matrix. This matrix consists of 3686 "unwell" patients and 362 "well" patients (4048 patients in total), that each answered 12 symptom-related questions (the "well" patients were asked about their

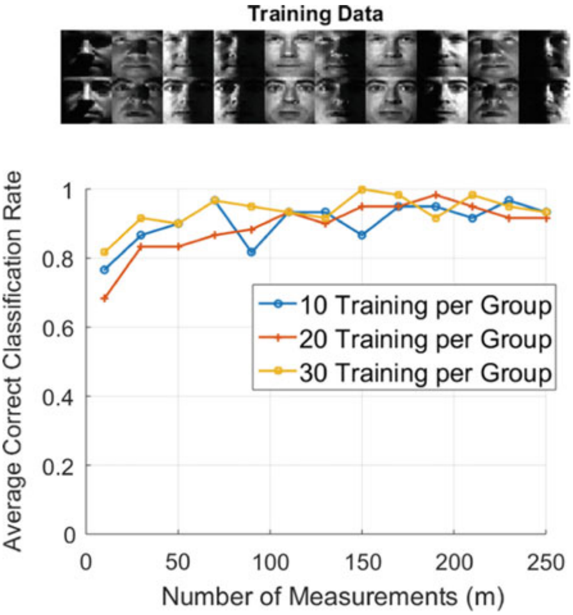


Fig. 7 Classification experiment using two individuals from the extended YaleB data set (top), $L = 6$, $n = 32 \times 32 = 1024$, 30 test points per group, and 30 trials of randomly generating A. Bottom: average correct classification rate versus m and for the indicated number of training points per class

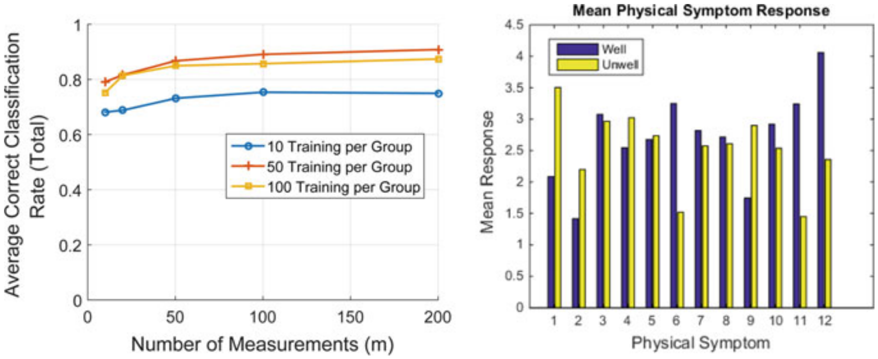


Fig. 8 Left: Results from classification approach on symptom data using 5 layers for various numbers of randomly selected training points (patients). Right: Means on the survey questions for these groups

worst symptoms while being sick). We randomly select a number of those patients from each group to serve as our training data, and the remaining as our test data. The left plot of Fig. 8 demonstrates the ability to accurately identify well versus unwell patients from the symptoms (current or past) that they report. Since the “well” patients were asked about their worst *prior* symptoms, one might ask whether it is simply the case that “well” patients showcase higher (or lower) symptom levels in general, making classification easy. However, the right plot of the figure demonstrates this is not the case, and that perhaps more intricate and complex symptom patterns are at work.

3 Hierarchical Classification

Next, we extend the classification approach described in the previous sections to the problem of hierarchical or multi-scale classification. In this setting, the class labels have additional structure, often taking the form of a tree. For example, in image classification problems, the data may contain images of inanimate and living objects. Then, within each of those classes the data may be further identified as images of vehicles and toys, or humans and animals. The data could then be further subdivided into classes of various animal types, and so on. Visualized as a tree, we view the children of each node as corresponding to its subclasses. Each data point in this case would have a label corresponding to a leaf of the tree, but also possesses the characteristics of all the labels of its ancestors. *Hierarchical classification* makes use of this information and structure between groups in classifying the data [23, 49]. Extensions of popular classification methods such as the support vector machine (SVM) to the hierarchical setting are not straightforward, and such approaches often decompose the problem into many subproblems leading to higher computational complexities [11, 52]. Here, we apply the simple classification method discussed in Sect. 2 to this hierarchical setting, and show that computational advantages are often possible. In particular, the method is likely to be particularly useful for hierarchical data in which certain subclasses of data are more or less difficult to classify than others.

We now describe the proposed adjustment for handling hierarchical classification, based on [39], where the labels possess some sort of tree structure. We use the same notation as for the methods described in previous sections. The key observation for the modification is that, if we know in advance that certain classes may require fewer levels for classification with sufficient accuracy, we may isolate these classes in an initial classification that uses fewer levels and then further classify among the remaining classes using more levels, as needed. This strategy leads to computational savings without sacrificing accuracy when some classes are more easily discerned from the others. Fortunately, this type of structure occurs naturally in many applications. For example, in medical brain imaging, it is typically much easier to classify patients with tumors than patients with various types of dementia [18, 25]. In cases like this, the method may utilize fewer levels for the easier classification steps. This approach is described formally in Algorithms 3 and 4.

Algorithm 3: Proposed adjustment for hierarchical classification (training).

```
Input: binary training data $Q$, training labels $b$, set of class groupings $S_c$ for each node $H_c$ in
       the tree of classifications $H$, number of levels $L = (L_1, \dots L_C)$ to be used in each
       classification.
for $H_c \in H$ do
    identify: $Q_c$, the submatrix of rows of $Q$ corresponding to training labels of $b$ contained
    in some set in $S_c$.
    define: $\tilde{b}$ as labels indicating to which set of $S_c$ a given row of $Q_c$ corresponds.
    train: a classifier as in Algorithm 1 with training data $Q_c$, labels $\tilde{b}$, number of groups
    $|S_c|$ and number of levels $L_c$ as input.
end
```

Algorithm 4: Proposed adjustment for hierarchical classification (testing).

```
Input: binary testing data $q$, set of class groupings $S_c$, learned parameters $r(\ell, i, t, g)$, $T_{\ell,i}$
       and $\Lambda_{l,i}$ for the classification associated to each node $H_c$ in the tree of classifications
       $H$, number of levels $L = (L_1, \dots L_C)$ to be used in each classification.
set: $H_c = H_1$, the root classification.
while $H_c$ is not null, do
    classify: $q$ into one of the sets contained in $S_c$, as in Algorithm 2, with learned
    parameters $r(\ell, i, t, g)$, $T_{l,i}$, $\Lambda_{l,i}$ from $H_c$.
    if $q$ is predicted to belong to a single class then
        set: $H_c$ to be null.
    else
        set: $H_c$ to the node corresponding to the predicted set of classes within $S_c$.
    end
end
```

3.1 Experimental Results

In this section, we showcase experiments from [39] that demonstrate the computational gains achieved by Algorithms 3 and 4 compared with direct classification into each individual group via “flat multiclass classification” as in Algorithms 1 and 2 (see Fig. 9). We first consider a simple two-dimensional example to aid in visualization; the data is shown in Fig. 10, where each color represents a different class from six classes in total. Since we expect classifying points from the red and yellow classes to be easier, we may use fewer levels than in classifying points as green, black, blue or cyan. Therefore, we first predict whether a test point is red or yellow versus green, black, blue or cyan using only one level. If the test point was predicted to be red or yellow, we then discern between these two classes again using only a single level. If the test point was predicted to be green, black, blue or cyan, we then predict among these classes by using varying numbers of levels. Accuracy and computational results are shown in Fig. 10 for varying numbers of measurements m. We see a significant reduction in computational cost using the hierarchical strategy without sacrificing accuracy. Note that the computational savings are realized for the test points predicted to belong to the red or yellow class, since classifying into these

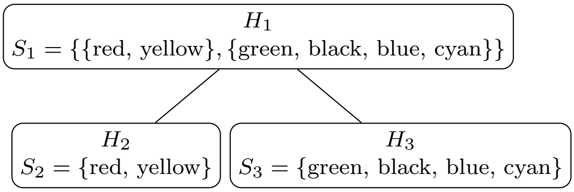

H_1
$S_1 = \{\{\text{red, yellow}\}, \{\text{green, black, blue, cyan}\}\}$
H_2
$S_2 = \{\text{red, yellow}\}$
H_3
$S_3 = \{\text{green, black, blue, cyan}\}$

Fig. 9 Hierarchical classification tree used to classify two-dimensional synthetic data as shown in Fig. 10

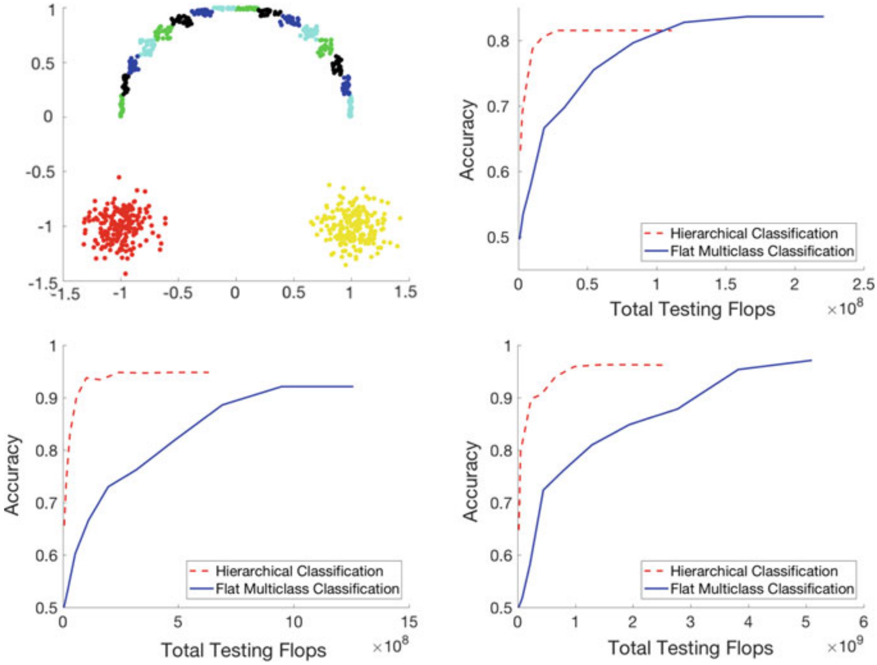


Fig. 10 For the data distributed as given in the upper left plot, where each color represents a different class, we classify test data either by flat multiclass classification or our proposed hierarchical classification strategy where the first classification discerns between red or yellow versus green, black, blue or cyan. Accuracy and testing flops required are given in the subsequent plots using varying numbers of levels and $m = 20$, 50 and 100 respectively. Results are averaged over 10 trials

groups requires fewer levels and thus fewer calculations. The computational savings of the hierarchical strategy are thus highly dependent on the distribution of the test data. In this experiment, we classify 200 test points from each of the red and yellow classes and 100 test points from each of the green, black, blue, and cyan classes, so that there are an equal number of test points from the "arc" and from the Gaussian clusters.

Although not inherently hierarchical in nature, we demonstrate that our hierarchical strategy can lead to computational savings on the MNIST data set of handwritten digits [36]. Consider the digits 1–5. Intuitively and in practice, the digit 1 tends to be easier to classify correctly than the other digits. For example, if we apply the multiclass classification from [41] to classify the digits 1–5 using 1000 training points for each class, 10 levels and testing on 200 training points from each class, we find that 98.5% of the 1s are classified correctly, whereas the overall accuracy of classifying the digits 1–5 was 89.2% (the accuracy for classifying digits 2–5 was 86.88%). Thus, it is reasonable to expect that fewer levels are required for sufficiently accurate classification of the 1s than are required to classify the remaining digits.

Considering the digits 1–5, we can thus induce hierarchical structure by first classifying into 1s (which tend to be easier) versus the other digits, followed by classification into the digits 2, 3, 4, and 5. Five levels are used for the first classification into 1s versus 2–5s and a varying number of levels (5–10) are used for the subsequent classification. We again see a reduction in the total testing flops required to achieve a given accuracy. Since this tree is fairly “shallow,” as expected the improvements are mild. We would expect a more significant reduction in computation via a hierarchical strategy for real data that has a larger and more imbalanced tree structure. Additionally, the test data includes an equal number of points corresponding to each digit, so we see computational savings for approximately 1/5 of the test points. If we expected the frequency of the digit 1 to be higher, we would expect the computational savings to be more significant as well.

4 Implementation Considerations

Here, we consider some implementation details and remarks for future work in this direction.

4.1 Parameter Selection

The key parameters the user must select in this simple classification approach are the number of measurements m and the number of levels L. The relationship between the number of measurements m and the performance of the method (see Corollaries 1 and 2) conforms with their analogous relationship in other settings like 1-bit compressed sensing and binary hashing (see e.g., [4, 22, 48]). Namely, increasing m exponentially improves the success probability of the method at hand. Henceforth, we focus here on the choice of levels L. We propose a simple scheme that uses the membership index function values on the training data to decide how many levels L are sufficient for accurate classification. This scheme can be viewed as a simple analog of cross-validation (Fig. 11).

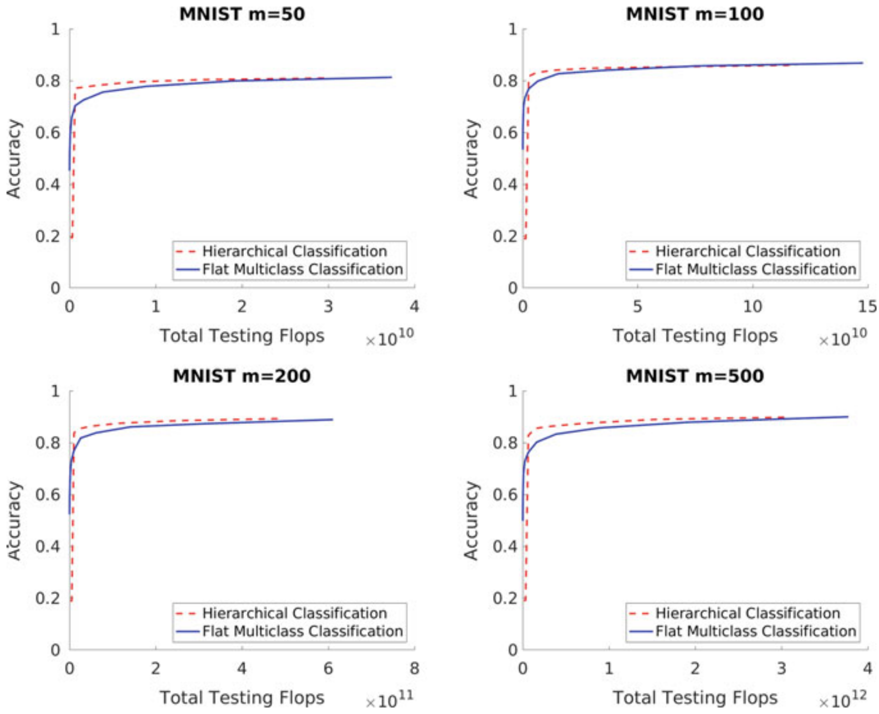


Fig. 11 Accuracy and testing flops required for flat multiclass classification versus the proposed hierarchical classification strategy in classifying digits 1–5 in the MNIST data set are given using m = 50, 100, 200, and 500, respectively. Results are averaged over 10 trials

Intuitively, the membership index values correspond to the level of confidence that a point belongs to a certain class. Thus ideally, for a fixed data point, we hope to see a single large membership value for one class and small values for the other classes. This motivates a scheme where examining the largest membership function value across classes, averaged over all the data, dictates an appropriate choice of L. More precisely, for a given level ℓ, one could consider running the testing method Algorithm 2 over all (or part of) the training data and computing the functions $\tilde{r}(g)$ for all represented classes g. Doing this at level ℓ for a data point with sign pattern t yields a value $\tilde{r}(g)$ for each class g, which we will now write as $\tilde{r}_{\ell,t}(g)$. We may then consider the average over all sign patterns at the level ℓ of the largest membership indices that is given by

$$\mu_\ell := \frac{1}{T}\sum_{t\in T}\max_g \tilde{r}_{\ell,t}(g),$$

where T is a set containing all represented sign patterns in the training data. We view large values of μ_ℓ as informing us that level ℓ is providing strong classification accuracy. Thus, if we view these values over various ℓ, we could stop using more levels once these values plateau or start decreasing. To verify this approach, we test this scheme on MNIST data for classifying 0–1 digits or 0–5 digits (with $p = 500$ and $m = 50$), see Figs. 12 and 13. Here, we notice that there appears to be a correlation between the level ℓ that yields maximal value μ_ℓ (right) and the point where using more levels does not lead to significant more accuracy (left).

Investigating this correlation quantitatively and theoretically will be an interesting direction for future work. For example, if we assume that the angle between any two classes is at least θ, we conjecture that after $L \leq O\left(\frac{1}{\theta^{n-1}}\right)$ levels (where n is the ambient dimension of the problem), the plateau begins, as adding more hyperplanes will create empty cones with high probability. Of course, a better bound should also depend on the total number m of hyperplanes out of which we select, and the number of test points p. Drawing such connections would be fruitful future directions of work.

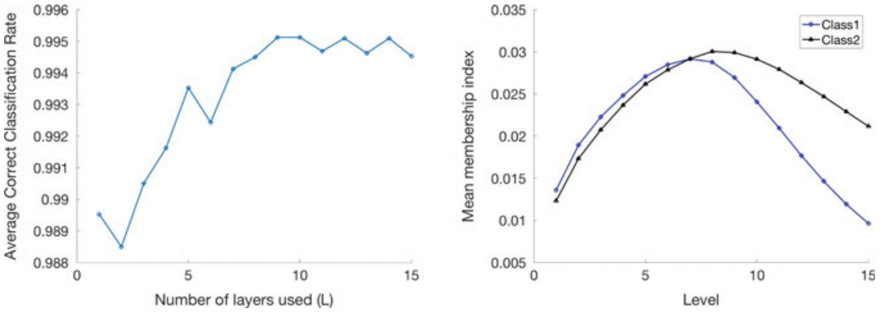


Fig. 12 MNIST 0–1 digits (single trial). Left: Average classification rate as function of levels. Right: Mean membership index function

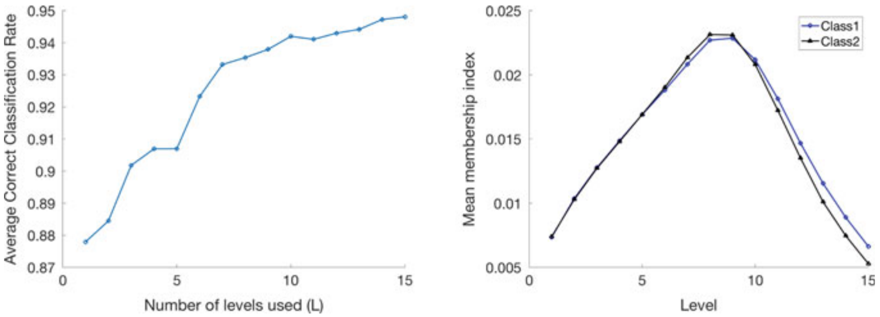


Fig. 13 MNIST 0 and 5 digits. Left: Average classification rate as function of levels. Right: Mean membership index function

4.2 Dynamic Hyperplane Selection

Another implementation concern and possible direction for future research is the optimal choice of hyperplanes on each layer. In the presented implementation of the classification algorithm, for each layer $\ell \in \{1, \dots, L\}$, the collections $\Lambda_{\ell,i}$ of hyperplanes are selected uniformly at random out of all possible ℓ-tuples. While this approach allows one to derive nice theoretical bounds such as Theorem 1, it might be beneficial for reconstruction if one instead chooses sets $\Lambda_{\ell,i}$ in a data-dependent way, so that hyperplanes in $\Lambda_{\ell,i}$ together provide a good separation of the training data. One might achieve this by using cross-validation or an approach similar to that described in Sect. 4.1, so that for each layer ℓ we reuse information obtained from the previous levels to decide which ℓ-tuples of hyperplanes could potentially allow for good class separation.

For instance, in the example described in Fig. 1 (right), we can see that for the blue and red hyperplanes, we have that in one half-space $\frac{2}{3}$ of all point are blue and $\frac{1}{3}$ are red, and in the other half-space all the points are red. Similarly, for the purple and green hyperplanes, we have that in both half-spaces, $\frac{1}{3}$ of all points are blue and $\frac{2}{3}$ are red. We can deduce that these pairs of points are "similar" in the sense that they divide the training data in similar ways. Thus it may be more beneficial to consider pairs of hyperplanes from different groups for the next level, that is {red, purple}, {blue, green}, {red, green}, and {blue, purple}. One can see that these pairs are indeed enough to separate clusters of training points. Alternatively, one could simply ignore hyperplane tuples that produce empty cones, which could happen frequently especially in high dimensions. Such dynamic selection of hyperplane tuples could lead to improved performance but perhaps more challenging analysis.

4.3 Efficient Representations

Next, we consider settings where the data in raw form is either not available or is too large to measure. Often, such data is instead available only by its adjacency graph, capturing distance measures between points. Such graphs arise naturally in many applications such as wireless communications, sensor networks and astronomy. Alternatively, we may wish to use such a representation to improve the classification accuracy. In this section, we demonstrate empirically that our approach is also robust to this type of data representation. In our first experiment, we use the MNIST 0–1 handwritten digit data but rather than measuring this data directly, we select a subset of training data and compute its adjacency matrix X where X_{ij} is the (Euclidean) distance between the ith and jth image. We then measure $Q = \text{sign}(AX)$ and proceed as usual. The results are shown in Fig. 14 (left), where actually we see an *improvement* in classification accuracy. We conjecture the improvement arises from the fact that

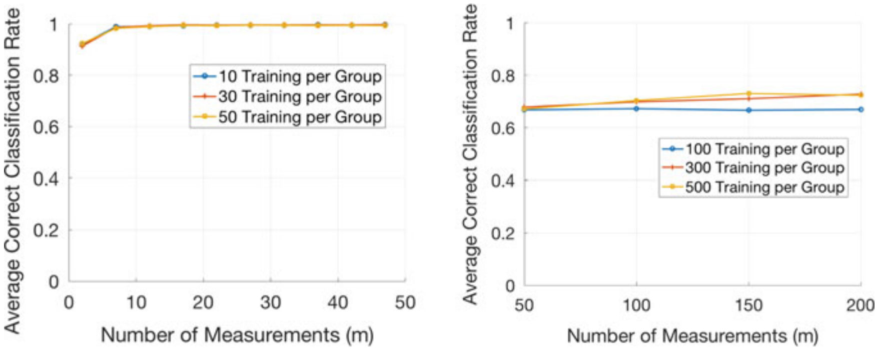


Fig. 14 Graph representations, average classification rate as function of levels. Left: MNIST 0–1 digits. Right: MNIST 0–9 digits

the adjacency columns are more linearly separable than the original raw data. Next, we use all ten digits of the MNIST data and create the adjacency matrix. Since the adjacency matrix scales with the number of training points, we are no longer free to use as many as we wish without bogging down computation. Thus, we are forced to use a smaller number of training points than in Fig. 6, and unsurprisingly, we see less accurate classification results, as shown in Fig. 14 (right). It would be interesting future work to study the geometry of such adjacency data and to develop an analogous analysis.

5 Conclusion

We have presented a simple classification method from [41] that can be applied to data represented in binary form. We have provided experimental results showcasing its classification accuracy on real and synthetic data as well as supporting theoretical analysis. In addition, we have demonstrated that the classification algorithm can be readily adapted to classify data in a hierarchical way that improves computational efficiency. In addition, we present some preliminary implementation modifications that can yield both computational and accuracy gains, and point out interesting directions for future work.

Acknowledgements Molitor and Needell were partially supported by NSF CAREER grant #1348721 and NSF BIGDATA #1740325. Saab was partially supported by the NSF under DMS-1517204.

References

1. N. Ailon, B. Chazelle, Approximate nearest neighbors and the fast Johnson-Lindenstrauss transform, in *Proceedings of the Thirty-Eighth Annual ACM Symposium on Theory of Computing* (ACM, 2006), pp. 557–563
2. D. Achlioptas, Database-friendly random projections: Johnson-Lindenstrauss with binary coins. J. Comput. Syst. Sci. **66**(4), 671–687 (2003)
3. A.M. Andrew, *An introduction to support vector machines and other kernel-based learning methods by Nello Christianini and John Shawe-Taylor* (Cambridge University Press, Cambridge, 2000), xiii+ pp. 189, ISBN 0-521-78019-5 (hbk, £ 27.50)
4. P. Boufounos, R. Baraniuk, 1-bit compressive sensing, in *Proceedings of IEEE Conference on Information, Science and Systems (CISS)* (Princeton, NJ, 2008)
5. R. Baraniuk, M. Davenport, R. DeVore, M. Wakin, The Johnson-Lindenstrauss lemma meets compressed sensing. Preprint **100**(1) (2006)
6. R. Baraniuk, S. Foucart, D. Needell, Y. Plan, M. Wootters, Exponential decay of reconstruction error from binary measurements of sparse signals. IEEE Trans. Inf. Theory **63**(6), 3368–3385 (2017)
7. A. Choromanska, K. Choromanski, M. Bojarski, T. Jebara, S. Kumar, Y. LeCun, Binary embeddings with structured hashed projections, in *Proceedings of The 33rd International Conference on Machine Learning* (2016), pp. 344–353
8. D. Cai, X. He, J. Han, Spectral regression for efficient regularized subspace learning, in *Proceedings of International Conference on Computer Vision (ICCV'07)* (2007)
9. D. Cai, X. He, Y. Hu, J. Han, T. Huang, Learning a spatially smooth subspace for face recognition, in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Machine Learning (CVPR'07)* (2007)
10. D. Cai, X. He, J. Han, H.-J. Zhang, Orthogonal laplacianfaces for face recognition. IEEE Trans. Image Process. **15**(11), 3608–3614 (2006)
11. S. Cheong, S.H. Oh, S.-Y. Lee, Support vector machines with binary tree architecture for multi-class classification. Neural Inf. Process. Lett. Rev. **2**(3), 47–51 (2004)
12. E. Candès, J. Romberg, T. Tao, Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. IEEE Trans. Inf. Theory **52**(2), 489–509 (2006)
13. E. Candès, J. Romberg, T. Tao, Stable signal recovery from incomplete and inaccurate measurements. Comm. Pure Appl. Math. **59**(8), 1207–1223 (2006)
14. N. Christianini, J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods* (Cambridge University Press, Cambridge, England, 2000)
15. S. Dasgupta, A. Gupta, An elementary proof of a theorem of Johnson and Lindenstrauss. Random Struct. Algorithms **22**(1), 60–65 (2003)
16. D. Donoho, Compressed sensing. IEEE Trans. Inf. Theory **52**(4), 1289–1306 (2006)
17. S. Dirksen, A. Stollenwerk, Fast binary embeddings with gaussian circulant matrices: improved bounds. arXiv preprint arXiv:1608.06498 (2016)
18. D. Duncan, T. Strohmer, Classification of alzheimer's disease using unsupervised diffusion component analysis. Math. Biosci. Eng. **13**, 1119–1130 (2016)
19. J. Fang, Y. Shen, H. Li, Z. Ren, Sparse signal recovery from one-bit quantized data: an iterative reweighted algorithm. Signal Process. **102**, 201–206 (2014)
20. Y. Gong, S. Lazebnik, A. Gordo, F. Perronnin, Iterative quantization: a procrustean approach to learning binary codes for large-scale image retrieval. IEEE Trans. Pattern Anal. Mach. Intell. **35**(12), 2916–2929 (2013)
21. S. Gopi, P. Netrapalli, P. Jain, A.V. Nori, One-bit compressed sensing: provable support and vector recovery. ICML **3**, 154–162 (2013)
22. A. Gupta, R. Nowak, B. Recht, Sample complexity for 1-bit compressed sensing and sparse classification, in *2010 IEEE International Symposium on Information Theory Proceedings (ISIT)* (IEEE, 2010), pp. 1553–1557
23. A.D. Gordon, A review of hierarchical classification. J. R. Stat. Soc. Ser. A Gen. 119–137 (1987)

24. M.A. Hearst, S.T. Dumais, E. Osuna, J. Platt, B. Scholkopf, Support vector machines. IEEE Intell. Syst. Appl. **13**(4), 18–28 (1998)
25. R. Higdon, N.L. Foster, R.A. Koeppe, C.S. DeCarli, W.J. Jagust, C.M. Clark, N.R. Barbas, S.E. Arnold, R.S. Turner, J.L. Heidebrink et al., A comparison of classification methods for differentiating fronto-temporal dementia from alzheimer's disease using FDG-PET imaging. Stat. Med. **23**(2), 315–326 (2004)
26. J. Hahn, S. Rosenkranz, A.M. Zoubir, Adaptive compressed classification for hyperspectral imagery, in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (IEEE, 2014), pp. 1020–1024
27. B. Hunter, T. Strohmer, T.E. Simos, G. Psihoyios, C. Tsitouras, Compressive spectral clustering, in *AIP Conference Proceedings*, vol. 1281 (AIP, 2010), pp. 1720–1722
28. X. He, S. Yan, Y. Hu, P. Niyogi, H.-J. Zhang, Face recognition using laplacianfaces. IEEE Trans. Pattern Anal. Mach. Intell. **27**(3), 328–340 (2005)
29. L. Jacques, K. Degraux, C. De Vleeschouwer, Quantized iterative hard thresholding: bridging 1-bit and high-resolution quantized compressed sensing. arXiv preprint arXiv:1305.1786 (2013)
30. W. Johnson, J. Lindenstrauss, Extensions of Lipschitz mappings into a Hilbert space, in *Proceedings of Conference in Modern Analysis and Probability* (New Haven, CT, 1982)
31. L. Jacques, J. Laska, P. Boufounos, R. Baraniuk, Robust 1-bit compressive sensing via binary stable embeddings of sparse vectors. IEEE Trans. Inf. Theory **59**(4), 2082–2102 (2013)
32. T. Joachims, Text categorization with support vector machines: learning with many relevant features, in *Machine Learning: ECML-98* (1998), pp. 137–142
33. A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in *Advances in Neural Information Processing Systems* (2012), pp. 1097–1105
34. K. Knudson, R. Saab, R. Ward, One-bit compressive sensing with norm estimation. IEEE Trans. Inf. Theory **62**(5), 2748–2758 (2016)
35. F. Krahmer, R. Ward, New and improved Johnson-Lindenstrauss embeddings via the restricted isometry property. SIAM J. Math. Anal. **43**(3), 1269–1281 (2011)
36. Y. LeCun, The MNIST database of handwritten digits. http://yann.lecun.com/exdb/mnist/
37. J.N. Laska, Z. Wen, W. Yin, R.G. Baraniuk, Trust, but verify: fast and accurate signal recovery from 1-bit compressive measurements. IEEE Trans. Signal Process. **59**(11), 5289–5301 (2011)
38. H. Li, Y. Yang, D. Chen, Z. Lin, Optimization algorithm inspired deep neural network structure design. arXiv preprint arXiv:1810.01638 (2018)
39. D. Molitor, D. Needell, Hierarchical classification using binary data (2018). Submitted
40. G.F. Montufar, R. Pascanu, K. Cho, Y. Bengio, On the number of linear regions of deep neural networks, in *Advances in Neural Information Processing Systems* (2014), pp. 2924–2932
41. D. Needell, R. Saab, T. Woolf, Simple classification using binary data. J. Mach. Learn. Res. (2017). Accepted
42. Y. Plan, R. Vershynin, One-bit compressed sensing by linear programming. Commun. Pure Appl. Math. **66**(8), 1275–1297 (2013)
43. Y. Plan, R. Vershynin, Robust 1-bit compressed sensing and sparse logistic regression: a convex programming approach. IEEE Trans. Inf. Theory **59**(1), 482–494 (2013)
44. Y. Plan, R. Vershynin, Dimension reduction by random hyperplane tessellations. Discret. Comput. Geom. **51**(2), 438–461 (2014)
45. O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein et al., Imagenet large scale visual recognition challenge. Int. J. Comput. Vision **115**(3), 211–252 (2015)
46. M. Rudelson, R. Vershynin, On sparse reconstruction from Fourier and Gaussian measurements. Comm. Pure Appl. Math. **61**(8), 1025–1171 (2008)
47. I. Steinwart, A. Christmann, *Support Vector Machines* (Springer Science & Business Media, 2008)
48. H.M. Shi, M. Case, X. Gu, S. Tu, D. Needell, Methods for quantized compressed sensing, in *Proceedings of Information Theory and Applications (ITA)* (2016)

49. C.N. Silla, A.A. Freitas, A survey of hierarchical classification across different application domains. Data Min. Knowl. Discov. **22**(1–2), 31–72 (2011)
50. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 1–9
51. K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
52. J. Weston, C. Watkins, Multi-class support vector machines. Technical Report (Citeseer, 1998)
53. X. Yi, C. Caravans, E. Price, Binary embedding: fundamental limits and fast algorithm (2015)
54. F.X. Yu, S. Kumar, Y. Gong, S.-F. Chang, Circulant binary embedding, in *International Conference on Machine Learning*, vol. 6 (2014), p. 7
55. M. Yan, Y. Yang, S. Osher, Robust 1-bit compressive sensing using adaptive outlier pursuit. IEEE Trans. Signal Process. **60**(7), 3868–3875 (2012)
56. Q.-S. Zhang, S.-C. Zhu, Visual interpretability for deep learning: a survey. Front. Inf. Technol. Electron. Eng. **19**(1), 27–39 (2018)

Generalization Error in Deep Learning

Daniel Jakubovitz, Raja Giryes and Miguel R. D. Rodrigues

Abstract Deep learning models have lately shown great performance in various fields such as computer vision, speech recognition, speech translation, and natural language processing. However, alongside their state-of-the-art performance, it is still generally unclear what is the source of their generalization ability. Thus, an important question is what makes deep neural networks able to generalize well from the training set to new data. In this chapter, we provide an overview of the existing theory and bounds for the characterization of the generalization error of deep neural networks, combining both classical and more recent theoretical and empirical results.

1 Introduction

Deep neural networks (DNNs) have lately shown tremendous empirical performance in many applications in various fields such as computer vision, speech recognition, speech translation, and natural language processing [1]. However, alongside their state-of-the-art performance in these domains, the source of their success and the reason for their being a powerful machine learning model remains elusive.

A deep neural network is a complex nonlinear model, whose training involves the solution of a non-convex optimization problem, usually solved with some variation of the stochastic gradient descent (SGD) algorithm. Even though convergence to a minimum with good performance is not guaranteed, it is often the case that the training of DNNs achieves both a small training error and good generalization results.

D. Jakubovitz · R. Giryes
School of Electrical Engineering, Tel Aviv University, Tel Aviv-Yafo, Israel
e-mail: danielshaij@mail.tau.ac.il

R. Giryes
e-mail: raja@tauex.tau.ac.il

M. R. D. Rodrigues (✉)
Department of Electronics and Electrical Engineering, University College London, London, UK
e-mail: m.rodrigues@ucl.ac.uk

H. Boche et al. (eds.), *Compressed Sensing and Its Applications*,
Applied and Numerical Harmonic Analysis,
https://doi.org/10.1007/978-3-319-73074-5_5

This chapter focuses on the characterization of the generalization abilities of neural networks. Indeed, there are various recent theoretical advances that aim to shed light on the performance of deep neural networks, borrowing from optimization theory, approximation theory, and related fields (e.g., see [2, 3] and others). Yet, due to space constraints, we concentrate here on over-viewing recent prominent approaches of statistical learning theory for understanding the generalization of deep neural networks.

The generalization error of a machine learning model is the difference between the empirical loss of the training set and the expected loss of a test set. In practice, it is measured by the difference between the error of the training data and the one of the test data. This measure represents the ability of the trained model (algorithm) to generalize well from the learning data to new unseen data. It is typically understood that good generalization is obtained when a machine learning model does not memorize the training data, but rather learns some underlying rule associated with the data generation process, thereby being able to extrapolate that rule from the training data to new unseen data and generalize well.

Therefore, the generalization error of DNNs has been the focus of extensive research, mainly aimed at better understanding the source of their capabilities and deriving key rules and relations between a network's architecture, the used optimization algorithm for training and the network's performance on a designated task. Bounds on the generalization error of deep learning models have also been obtained, typically under specific constraints (e.g., a bound for a two-layer neural network with ReLU activations). Recent research also focuses on new techniques for reducing a network's generalization error, increasing its stability to input data variability and increasing its robustness.

The capabilities of deep learning models are often examined under notions of expressivity and capacity: their ability to learn a function of some complexity from a given set of examples. It has been shown that deep learning models are capable of high expressivity, and are hence able to learn any function under certain architectural constraints. However, classical measures of machine learning model expressivity (such as Vapnik–Chervonenkis (VC) dimension [4], Rademacher complexity [5], etc.), which successfully characterize the behavior of many machine learning algorithms, fail to explain the generalization abilities of DNNs. Since DNNs are typically overparameterized models with substantially less training data than model parameters, they are expected to overfit the training data and obtain poor generalization as a consequence [6]. However, this is not the case in practice. Thus, a specific line of work has been dedicated to study the generalization of these networks.

Several different theories have been suggested to explain what makes a DNN generalize well. As a result, several different bounds for the generalization error of DNNs have been proposed along with techniques for obtaining better generalization in practice. These rely on measures such as the PAC-Bayes theory [7–9], algorithm stability [10], algorithm robustness [11] and more.

In the following sections, we survey the theoretical foundations of the generalization capabilities of machine learning models with a specific emphasis on deep neural networks, the corresponding bounds on their generalization error, and several insights and techniques for reducing this error in practice. Namely, we review both classical and more recent theoretical works and empirical findings related to the generalization capabilities of machine learning algorithms, and specifically deep neural networks.

2 The Learning Problem

Machine learning is a field which employs statistical models in order to *learn* how to perform a designated task without having to explicitly program for it. It is closely related to and inspired by other domains in applied mathematics, computer science, and engineering such as optimization, data mining, pattern recognition, statistics and more. Accordingly, machine learning models and methods are inspired by the prominent techniques, models and algorithms of these fields [1, 12].

Core to the field of machine learning is the learning (training) process, in which an algorithm is given a training dataset in order to learn how to perform a desired task by learning some underlying rule associated with the data generation process. After the learning phase is done, a good algorithm is expected to perform its task well on unseen data drawn from the same underlying rule. This phase is commonly referred to as the test phase, in which an algorithm performs its designated task on new data. In general, machine learning can be divided into two categories. The first category is supervised learning, in which there is a ground truth value (label) for each data sample. This ground truth value is supplied to the algorithm as part of its training dataset, and is expected to be correctly predicted by the algorithm during the test phase for new unseen data. The second category is unsupervised learning, in which there are no ground truth labels that characterize the data, and it is up to the algorithm itself to characterize the data correctly and efficiently in order to perform its task.

Some of the most prominent machine learning algorithms are the Support Vector Machine (SVM), K-Nearest Neighbors (K-NN), K-Means, decision trees, deep neural networks, etc. [12]. These algorithms are used to perform a variety of different tasks such as regression, classification, clustering, and more. Deep neural networks, which are the subject of this chapter, are a particular model (algorithm) that has attracted much interest in the past several years due to its astonishing performance and generalization capabilities in a variety of tasks [1].

The following notation is used throughout this chapter. The input space of a learning algorithm is the D-dimensional subspace $\mathcal{X} \subseteq \mathbb{R}^D$ and $x \in \mathcal{X}$ is an input sample to the algorithm. The output space is the K-dimensional subspace $\mathcal{Y} \subseteq \mathbb{R}^K$. The label of the input sample x is $y \in \mathcal{Y}$. The sample set is denoted as $\mathcal{Z}$, where $\mathcal{Z} = \mathcal{Y} \times \mathcal{X}$.

We will concentrate predominantly on classification tasks. Consequently, K corresponds to the number of possible classes and $k^* \in \{1, \dots, K\}$ is the correct class of the input sample x. Accordingly, the label vector $y \in \mathcal{Y}$ is a one-hot vector, meaning all its elements are equal to zero except for the element in the k^*th index which is equal to one. The function f_W is the learned function of a model with parameters $W \in \mathcal{W}$ (where $\mathcal{W}$ is the parameter space), i.e., $f_W : \mathbb{R}^D \to \mathbb{R}^K$. The kth value of the vector $f_W(x)$ is denoted by $f_W(x)[k]$. In most cases we omit the W symbolizing the network's parameters for convenience.

We assume a training dataset has N training examples, such that the set $\mathbf{s}_N = \{s_i | s_i \in \mathcal{Z}\}_{i=1}^N = \{(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}\}_{i=1}^N$ is the algorithm's training set. The samples are independently drawn from a probability distribution $\mathcal{D}$. The set $\mathbf{t}_{N_{test}} = \{t_i | t_i \in \mathcal{Z}\}_{i=1}^{N_{test}} = \{(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}\}_{i=1}^{N_{test}}$ is the algorithm's test set, which consists of N_{test} test examples. The hypothesis set, which consists of all possible functions f_W, is denoted as $\mathcal{H}$. Therefore, a learning algorithm $\mathcal{A}$ is a mapping from $\mathcal{Z}^N$ to $\mathcal{H}$, i.e., $\mathcal{A} : \mathcal{Z}^N \to \mathcal{H}$.

The loss function, which measures the discrepancy between the true label y and the algorithm's estimated label $f(x)$ is denoted by $\ell\,(y, f(x))$. In some cases, when referring to the loss of a specific learning algorithm $\mathcal{A}$ trained on the set $\mathbf{s}_N$ and evaluated on the sample s, we denote $\ell(\mathcal{A}_{\mathbf{s}_N}, s)$ instead.

For a general loss function ℓ, an algorithm's empirical loss on the training set (train loss) $\mathbf{s}_N$ is

$$\ell_{emp}(f, \mathbf{s}_N) = \ell_{emp}(\mathcal{A}_{\mathbf{s}_N}) \triangleq \frac{1}{N} \sum_{i=1}^{N} \ell\,(y_i, f(x_i))\,, \quad \{(x_i, y_i)\}_{i=1}^N \in \mathbf{s}_N,$$

and the expected loss of the algorithm is

$$\ell_{exp}(f) = \ell_{exp}(\mathcal{A}_{\mathbf{s}_N}) \triangleq \mathbb{E}_{(x,y)\sim\mathcal{D}}\,[\ell(y, f(x))]\,.$$

Accordingly, an algorithm's *generalization error* is given by

$$GE(f, \mathbf{s}_N) \triangleq \left|\ell_{emp}(f, \mathbf{s}_N) - \ell_{exp}(f)\right|.$$

The empirical test loss is often used to approximate the expected loss since the distribution $\mathcal{D}$ is unknown to the learning algorithm. The test loss of an algorithm is given by

$$\ell_{test}(f, \mathbf{t}_{N_{test}}) \triangleq \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} \ell(y_i, f(x_i)), \quad \{(x_i, y_i)\}_{i=1}^{N_{test}} \in \mathbf{t}_{N_{test}},$$

and the corresponding approximation of the *generalization error* is given by

$$GE(f, \mathbf{s}_N, \mathbf{t}_{N_{test}}) \triangleq \left|\ell_{emp}(f, \mathbf{s}_N) - \ell_{test}(f, \mathbf{t}_{N_{test}})\right|.$$

The output classification margin γ of a data sample x is defined by the difference between the value of the correct class and the maximal value over all other classes: $\gamma = f(x)[k^*] - \max_{k \neq k^*} f(x)[k]$, where as mentioned earlier, k^* corresponds to the index associated with the correct class of the data sample x, i.e., $y[k^*] = 1$ and $y[k] = 0\ \forall k \neq k^*$. The margin loss is defined as follows.

The empirical margin loss for an output margin γ is

$$\ell_{emp,\gamma}(f, \mathbf{s}_N) \triangleq \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}\left\{ f(x_i)[k_i^*] - \max_{k \neq k_i^*} f(x_i)[k] \leq \gamma \right\},\ \{(x_i, y_i)\}_{i=1}^{N} \in \mathbf{s}_N,$$

where $\mathbb{1}$ signifies the indicator function that gets the value one if the inequality holds and the value zero otherwise. The expected margin loss for an output margin γ is

$$\ell_{exp,\gamma}(f) \triangleq Pr_{(x,y)\sim\mathcal{D}}\left[f(x)[k^*] - \max_{k \neq k^*} f(x)[k] \leq \gamma \right].$$

We denote the Frobenius, ℓ_1, ℓ_2 and ℓ_∞ norms by $||\cdot||_F$, $||\cdot||_1$, $||\cdot||_2$ and $||\cdot||_\infty$ respectively.

The training of machine learning algorithms relies on the Empirical Risk Minimization (ERM) principle. Since a learning algorithm only has access to a finite amount of samples drawn from the probability distribution $\mathcal{D}$, which is unknown, it aims at minimizing the *empirical risk* represented by the training loss $\ell_{emp}(\mathcal{A}_{\mathbf{s}_N})$. This practice can be suboptimal, as it is subject to the risk of overfitting the specific training samples. The term "over-fitting" refers to a phenomenon in which a learning algorithm fits the specifics of the training samples "too well", thereby representing the underlying distribution $\mathcal{D}$ poorly.

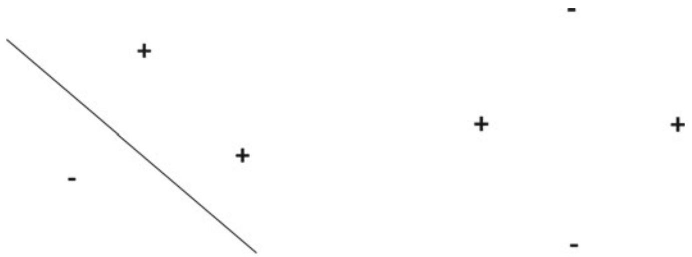
Throughout this chapter the notion of model (algorithm) *capacity* is used. This is a general term that relates to the capability of a model to represent functions of a certain complexity. It is evaluated in different ways in different contexts. A formal and accurate definition is given where relevant along this chapter.

Two classical metrics which are used to evaluate the capacity (or expressivity) of learning algorithms are the VC-dimension [4] and the Rademacher complexity [5]. The VC-dimension measures the classification capacity of a set of learned functions.

Definition 1 (*VC-dimension*) A classification function f with parameters W shatters a set of data samples $\{x_i\}_{i=1}^{N}$ if for all possible corresponding labels $\{y_i\}_{i=1}^{N}$ there exist parameters W such that f makes no classification errors on this set. The VC-dimension of f is the maximum amount of data samples N such that f shatters the set $\{x_i\}_{i=1}^{N}$. If no such maximal value exists then the VC-dimension is equal to infinity.

To gain intuition as to the meaning of the VC-dimension, let us consider the following example.

Example 1 Let us consider a linear function in the space $\mathbb{R}^2$. The function $\alpha_1 x_1 + \alpha_2 x_2 + b = 0$, which is parameterized by $W = (\alpha_1, \alpha_2, b) \in \mathbb{R}^3$, defines a classification decision for any sample $x = (x_1, x_2) \in \mathbb{R}^2$ according to the following rule:



Three samples in $\mathbb{R}^2$ correctly classified by a linear function.

Four samples in $\mathbb{R}^2$ which cannot be correctly classified by a linear function.

Fig. 1 Linear classification of samples in $\mathbb{R}^2$

$$f(x) = \begin{cases} +1, & \text{if } \alpha_1 x_1 + \alpha_2 x_2 + b \geq 0 \\ -1, & \text{otherwise} \end{cases}$$

This means that any sample above or on the line is classified as positive $(+1)$, whereas any sample under the line is classified as negative (-1). Let us define the hypothesis set:

$$\mathcal{H} = \{f_W | W = (\alpha_1, \alpha_2, b) \in \mathbb{R}^3\},$$

then $\text{VCdim}(\mathcal{H}) = 3$.

Proof sketch. Note that in this case, any three samples in $\mathbb{R}^2$ (which are not colinear) can be shattered by a linear classifier. However, four samples in $\mathbb{R}^2$ can be easily chosen such that no linear function can represent the correct classification rule. See Fig. 1 for an illustration of these cases.

The Rademacher complexity measures the richness of a set of functions with respect to some probability distribution. Essentially, it measures the ability of a set of functions to fit random ±1 labels.

Definition 2 (*Rademacher complexity*) Given a dataset $\mathbf{s_x} = \{(x_i)\}_{i=1}^N$, and a hypothesis set of functions $\mathcal{H}$, the empirical Rademacher complexity of $\mathcal{H}$ given $\mathbf{s_x}$ is

$$\mathcal{R}_N(\mathcal{H}) = \mathbb{E}_\sigma \left[\sup_{h \in \mathcal{H}} \frac{1}{N} \sum_{i=1}^{N} \sigma_i h(x_i) \right], \tag{1}$$

where $\sigma_i \in \{\pm 1\}$, $i = 1, \ldots, N$ are independent and identically distributed uniform random variables, i.e., $Pr(\sigma_i = 1) = Pr(\sigma_i = -1) = \frac{1}{2}, i = 1, \ldots, N$.

Note that since $h(x_i) \in \{\pm 1\}$, if $\sigma_i h(x_i) = 1$ the classification is correct and if $\sigma_i h(x_i) = -1$ the classification is wrong, and therefore we seek to maximize the sum in (1). To gain intuition as to the meaning of the Rademacher complexity, let us consider the following example of linear classifiers.

Example 2 Let $\mathcal{H}$ be the class of linear classifiers with an ℓ_2 norm of the weights bounded by α: $\{\{w^T x_i\}_{i=1}^N, ||w||_2 \le \alpha\}$. Let us assume that the ℓ_2 norm of the samples in the dataset $\mathbf{s_x} = \{x_i\}_{i=1}^N$ is upper bounded by β, i.e., $||x_i||_2 \le \beta, \forall i = 1, \ldots, N$. Then the Rademacher complexity of $\mathcal{H}$ is upper bounded as

$$\mathcal{R}_N(\mathcal{H}) \le \frac{\alpha\beta}{\sqrt{N}}.$$

Proof According to the definition of the Rademacher complexity,

$$\mathcal{R}_N(\mathcal{H}) = \mathbb{E}_\sigma \left[\sup_{w:||w||_2 \le \alpha} \frac{1}{N} \sum_{i=1}^N \sigma_i w^T x_i \right]$$

$$= \frac{1}{N} \mathbb{E}_\sigma \left[\sup_{w:||w||_2 \le \alpha} w^T \left(\sum_{i=1}^N \sigma_i x_i \right) \right] \le \frac{\alpha}{N} \mathbb{E}_\sigma \left[|| \sum_{i=1}^N \sigma_i x_i ||_2 \right]$$

$$\le \frac{\alpha}{N} \sqrt{\mathbb{E}_\sigma \left[|| \sum_{i=1}^N \sigma_i x_i ||_2^2 \right]} = \frac{\alpha}{N} \sqrt{\mathbb{E}_\sigma \left[|| \sum_{i=1}^N x_i ||_2^2 \right]} \tag{2}$$

$$= \frac{\alpha\beta}{\sqrt{N}},$$

where in (2) we used Jensen's inequality.

As described throughout this chapter, bounds on these measures of complexity of a learned function are generally unable to explain the generalization capabilities of deep neural networks, and are therefore more suited for the analysis of classical, less complex machine learning algorithms such as the support vector machines (SVM), K-NN, and others [12, 13].

Another commonly used framework for the analysis of machine learning algorithms is the PAC-Bayes theorem (Probably Approximately Correct), which is used to bound the generalization error of stochastic classifiers [7–9]. The PAC-Bayes framework provides a generalization bound which relates a prior distribution P, postulated before any data was seen, and a posterior distribution Q, which depends on the data (i.e., the training set). Unlike the VC-dimension and Rademacher complexity, the PAC-Bayes framework refers to the *distribution* of the hypothesis set of learned functions rather than a specific classification function. One should keep in mind that this is a general framework for the analysis of machine learning algorithms, from which several different bounds and mathematical formulations have been derived. We present hereafter one of its core theorems.

Theorem 1 (PAC-Bayes theorem) *Let $\mathcal{D}$ be a distribution over $\mathcal{Y} \times \mathcal{X}$ from which examples are drawn independently. Let P and Q denote probability distributions over the hypothesis set of classifiers $\mathcal{H}$. In addition, let $Err_{\mathcal{D}}(Q) = \mathbb{E}_{f\sim Q}[Pr_{(x,y)\sim\mathcal{D}}(y \neq f(x))]$ be the expected test probability of error and let $Err_{\mathbf{s}}(Q) = \frac{1}{N}\sum_{i=1}^{N}\mathbb{E}_{f\sim Q}$ $[\mathbb{1}\{y_i \neq f(x_i)\}]$, $\{(x_i, y_i)\}_{i=1}^{N} \in \mathbf{s}_N$ be the expected empirical training probability of error of the stochastic classifier Q over the training set $\mathbf{s}_N$ sampled from $\mathcal{D}$. Then for any Q and any $\delta \in (0, 1)$, we have with probability at least $1-\delta$ that over N randomly drawn training samples,*

$$KL\left(Err_{\mathbf{s}}(Q)||Err_{\mathcal{D}}(Q)\right) \leq \frac{KL(Q||P) + \ln(\frac{N+1}{\delta})}{N}$$

holds for all distributions P.

In the above, $KL(\cdot||\cdot)$ denotes the Kullback–Leibler divergence between two probability distributions and $\mathbb{1}$ denotes the indicator function which gets the value one if the inequality holds and zero otherwise. Note that P is an a priori distribution and Q is a posterior distribution given the training dataset $\mathbf{s}_N$.

The notion of algorithm robustness was introduced in [11]. A learning algorithm is said to be *robust* if for a training sample and a test sample that are close to each other, a similar performance is achieved. The following is the formal definition of a robust learning algorithm.

Definition 3 (*Robustness*) Algorithm $\mathcal{A}$ is $(K, \epsilon(\mathbf{s}))$ robust if $\mathcal{Z}$ can be partitioned into K disjoint sets, denoted as $\{C_i\}_{i=1}^{K}$, such that $\forall s \in \mathbf{s}$,

$$s, z \in C_i, \Rightarrow |\ell(\mathcal{A}_{\mathbf{s}}, s) - \ell(\mathcal{A}_{\mathbf{s}}, z)| \leq \epsilon(\mathbf{s}). \tag{3}$$

Note that $\ell(\mathcal{A}_{\mathbf{s}}, s)$ is the loss on the sample s of the algorithm $\mathcal{A}_{\mathbf{s}}$ which was trained on the set $\mathbf{s}$. A weaker definition of robustness, pseudo-robustness, is also useful for the analysis of the generalization error of learning algorithms, and is given in Sect. 4.5.

The notion of *sharpness* of the obtained solution to the minimization problem of the training of DNNs, i.e., the minimizer of the training loss, has lately become key in the analysis of the generalization capabilities of DNNs. Though several different definitions exist, we rely on the definition from [14] which is in wide use. Formally, the sharpness of the obtained minimizer is determined by the eigenvalues of the Hessian matrix $\nabla^2\ell_{emp}(\mathcal{A}_{\mathbf{s}_N})$ evaluated at the minimizer. However, since the computation of the Hessian matrix of DNNs is computationally expensive, an alternative measure is used. This measure relies on the evaluation of the maximal value of $\nabla^2\ell_{emp}(\mathcal{A}_{\mathbf{s}_N})$ in the environment of the examined solution. The maximization is done both on the entire input space $\mathbb{R}^D$ and on P-dimensional random manifolds, using a random matrix $A_{D\times P}$.

Definition 4 (*Sharpness*) Let C_ϵ denote a box around the solution over which the maximization of ℓ is performed. The constraint C_ϵ is defined by

$$C_\epsilon = \left\{z \in \mathbb{R}^P : -\epsilon(|(A^\dagger x)_i| + 1) \le z_i \le \epsilon(|(A^\dagger x)_i| + 1) \quad \forall i \in \{1, \dots, P\}\right\}$$

where $A^\dagger$ denotes the pseudoinverse of A. The value of $\epsilon > 0$ controls the size of the box. Given $x \in \mathbb{R}^D$, $\epsilon > 0$ and $A \in \mathbb{R}^{D\times P}$, the (C_ϵ, A)-sharpness of ℓ at x is defined by

$$\phi_{x,\ell}(\epsilon, A) \triangleq \frac{(\max_{y\in C_\epsilon} \ell(x + Ay)) - \ell(x)}{1 + \ell(x)} \times 100.$$

In a recent work [15], a compression- based approach is used to derive bounds on the generalization error of a classifier. A compressible classifier is defined as follows.

Definition 5 (*Compressibility*) Let f be a classifier and $G_{\mathcal{W}} = \{g_W | W \in \mathcal{W}\}$ be a class of classifiers such that g_W is uniquely determined by W. Then f is $(\gamma, \mathbf{s})$-compressible via $G_{\mathcal{W}}$ with an output margin $\gamma > 0$, if there exists $W \in \mathcal{W}$ such that for any sample in the dataset $x \in \mathbf{s}$ we have for all k

$$|f(x)[k] - g_W(x)[k]| \le \gamma. \tag{4}$$

Note that $f(x)[k]$ is the kth entry in the K-dimensional vector $f(x)$.

3 Deep Neural Networks

In this section, we give the definition of a deep neural network and explain several aspects of its architecture. Readers familiar with deep neural networks can skip directly to Sect. 4.

Deep neural networks, often abbreviated as simply “networks”, are a machine learning model which generally consists of several concatenated layers. The network processes the input data by propagating it through its layers for the purpose of performing a certain task. When a network consists of many layers it is commonly referred to as a “deep neural network”. A conventional feedforward neural network, which is the focus of the works we survey hereafter, has the following structure. It consists of L layers, where the first $L - 1$ layers are referred to as “hidden layers” and the Lth layer represents the network’s output. Each layer in the network consists of several neurons (nodes). An illustration of a neural network is given in Fig. 2.

Feedforward neural networks are networks in which the data propagates in a single direction, as opposed to other neural network models such as Recurrent Neural Networks (RNNs) in which the network connections form internal cycles. Though many different variations of feedforward neural networks exist, classically they either have fully connected layers or convolutional layers. A feedforward neural network with at least one convolutional layer, in which at least one convolution kernel is used, is referred to as a Convolutional Neural Network (CNN). In standard fully connected networks, every neuron in every layer is connected to each neuron in the previous

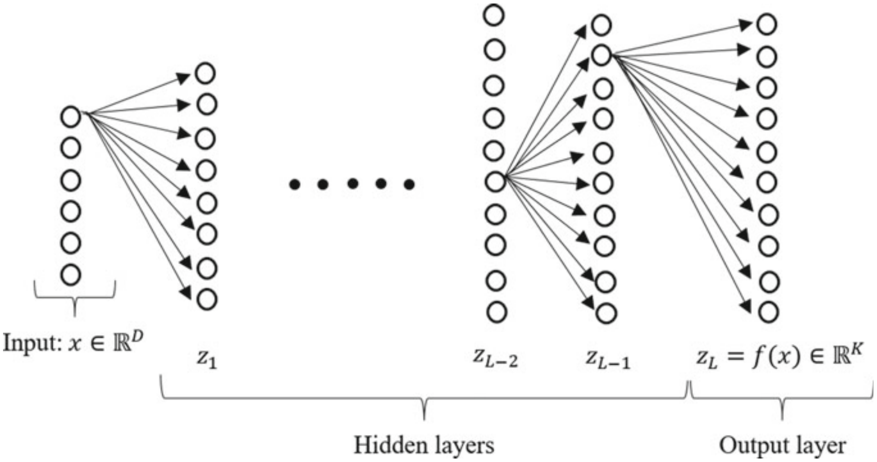


Fig. 2 A deep neural network. Some of the connections are omitted for simplicity

Table 1 Nonlinear activation functions

Name	Function: $\phi(x)$	Derivative: $\frac{d\phi(x)}{dx}$	Function output range
ReLU	max{0,x}	1 if $x > 0$; 0 if $x \leq 0$	$[0, \infty)$
Sigmoid	$\frac{1}{1+e^{-x}}$	$\phi(x)(1-\phi(x)) = \frac{e^{-x}}{(1+e^{-x})^2}$	(0,1)
Hyperbolic tangent	$\frac{e^x-e^{-x}}{e^x+e^{-x}}$	$1-\phi(x)^2 = 1-\left(\frac{e^x-e^{-x}}{e^x+e^{-x}}\right)^2$	$(-1,1)$

layer. Such a connection is mathematically defined as a linear transformation using a weight matrix followed by the addition of a bias term, which is then followed by a nonlinear activation function. The most commonly used activation functions are the Rectified Linear Unit (ReLU), the sigmoid function and the hyperbolic tangent (tanh) function. These activation functions are described in Table 1. There are other nonlinearities that can be applied to a layer's output, for example, pooling which decreases the layer's dimensions by aggregating information.

We use the index $l = 1, \ldots, L$ to denote a specific layer in the network, and h_l to denote the amount of neurons in the lth layer of the network. Accordingly, $h_0 = D$ and $h_L = K$ represent the network's input and output dimensions, respectively. The h_l-dimensional output of the lth layer in the network is represented by the vector z_l. The output of the last layer, z_L is also denoted by $f \triangleq z_L$, representing the network's output. The weight matrix of the lth layer of a network is denoted by $W_l \in \mathbb{R}^{h_l \times h_{l-1}}$. We denote the element in the ith row and the jth column in the weight matrix of the lth layer in the network by w_{ij}^l. The bias vector of the lth layer is denoted by $b_l \in \mathbb{R}^{h_l}$. In addition, $vec\left(\{W_l\}_{l=1}^L\right)$ is the column-stack vector representation of all of the network's weights. The activation function applied to every neuron in this layer is denoted by ϕ_l, which is applied element-wise when its input is a vector. Consequently, the relation between two consecutive layers in a fully connected DNN is given by

$$z_l = \phi_l \left(W_l z_{l-1} + b_l \right).$$

In most cases, the same nonlinear activation function is chosen for all the layers of the network.

In classification tasks, a neural network can be used to classify an input to one of K discrete classes, denoted using the index $k = 1, \dots, K$. A common choice for the last layer of a network performing a classification task is the softmax layer, which transforms the output range to be between 0 and 1:

$$f[k] = z_L[k] = \text{softmax}\{z_{L-1}[k]\} = \frac{e^{z_{L-1}[k]}}{\sum_{j=1}^{h_{L-1}} e^{z_{L-1}[j]}}.$$

The predicted class k^* for an input x is determined by the index of the maximal value in the output function obtained for this input sample, i.e.,

$$k^* = \text{argmax}_k f[k].$$

Since for any $k = 1, \dots, K$ the output range is $f[k] \in (0, 1)$, the elements of $f \in \mathbb{R}^K$ (the network's output) are usually interpreted as probabilities assigned by the network to the corresponding class labels, and accordingly the class with the highest probability is chosen as the predicted class for a specific input. The usage of the softmax layer is usually coupled with the cross-entropy loss function defined by

$$\ell(y, f(x)) = -\sum_{k=1}^{K} y[k] \log \left(f(x)[k] \right) = -\log \left(f(x)[k^*] \right),$$

where k^* is the index of the correct class of the input x.

The training of a neural network involves the solution of an optimization problem which encapsulates the discrepancy between the true labels and the estimated labels of the training dataset, computed using a loss function.

Typically used loss functions are the cross-entropy loss defined above, the squared error loss (using the ℓ_2 norm) defined by

$$\ell(y, f(x)) = ||y - f(x)||_2^2 = \sum_{k=1}^{K} \left(y[k] - f(x)[k] \right)^2,$$

the absolute error loss (using the ℓ_1 norm) defined by

$$\ell(y, f(x)) = ||y - f(x)||_1 = \sum_{k=1}^{K} |y[k] - f(x)[k]| ,$$

etc. In many cases, a closed-form solution is unobtainable, whereas in other cases it is too computationally demanding. Therefore, the optimization problem is usually solved using some variant of the gradient descent algorithm, which is an iterative algorithm. In every iteration, a step is taken in the direction of the negative gradient vector of the loss function, i.e., the direction of the steepest descent, and the model parameters are updated accordingly. The size of the taken step is tuned using a scalar hyperparameter, most commonly referred to as the "learning rate". The update of the model parameters in every iteration is given by

$$W_n = W_{n-1} - \alpha \nabla_W \ell_{emp}(f, \mathbf{s}), \tag{5}$$

where n represents the training iteration and α represents the learning rate. In most cases, in every iteration only a subset of the training dataset is used to compute the gradient of the loss function. This subset is commonly referred to as a training "mini-batch". Note that when the entire training dataset is used to evaluate the gradient of the loss, the optimization algorithm is usually referred to as "Batch Gradient Descent", whereas when a subset of the training dataset is used the optimization algorithm is referred to as "Stochastic Gradient Descent" (SGD). The usage of training mini-batches is computationally beneficial as it requires the usage of less data in each training iteration. It has other advantages as well, as will be detailed later in this chapter.

The computation of the gradient of the loss function with respect to the model parameters, which is necessary to perform an optimization step, is typically a costly operation as the relation between the input and the output of a DNN is quite complex and cannot be expressed in a closed-form expression. In most implementations this computation is based on the "back-propagation" algorithm, which is based on the chain rule for the derivation of the composition of functions [1]. This algorithm essentially computes the product of the partial derivatives along the different layers of the networks, propagating from the network's output to its input.

When training a neural network, and other machine learning algorithms as well, it is common practice to incorporate the usage of regularization techniques. This is effectively equivalent to making a prior assumption on the model itself or its input data. Using regularization techniques introduces several benefits. First, these techniques discourage the learned algorithm from overfitting the training data, i.e., they encourage the learning algorithm to learn the underlying rule of the training data rather than memorize the specifics of the training dataset itself. This purpose is achieved by essentially penalizing the learned algorithm for being too complex and "artificially" fitting the specifics of the training data. Second, regularization techniques promote the stability of the learned algorithm, in the sense that a small change to the input would not incur a large change to the output [13].

Regularization techniques can generally be categorized as either explicit or implicit. Explicit regularization techniques traditionally incorporate an additional loss function into the training objective function directly aimed at adapting the model to the assumed prior. When an additional regularization loss is used, the balance between the regularization term and the objective loss function is tuned using a multiplicative scalar hyperparameter. Commonly used explicit regularization techniques include the incorporation of an additional loss in the form of the norm of the model weights (parameters), the usage of dropout, in which any neuron in the network is zeroed during training with a certain probability, and more. Implicit regularization techniques have a more indirect influence on the learned function, and include a variety of techniques such as early stopping, data augmentation, and also model choices such as the used network architecture and optimization algorithm [1].

4 Generalization Error in Deep Learning

An important open question in machine learning research is what is the source of the generalization capabilities of deep neural networks. A better understanding of what affects this generalization error is essential toward obtaining reliable and robust deep neural network architectures with good performance. Throughout the following subsections we review different theories, insights and empirical results that shed light on this topic. Several lines of work aim at bounding the generalization error of neural networks, whereas others seek for complexity measures that correlate to this error and explain what affects it. In addition, we review both works that characterize different aspects of the training phase, ranging from the size of the training mini-batch to the number of training iterations, and works that characterize the solution of the training optimization problem. These works represent the most prominent lines of research in this field.

4.1 Understanding Deep Learning Requires Rethinking Generalization

As described in Sect. 3, regularization techniques such as weight decay and dropout have been shown to improve the generalization capabilities of machine learning algorithms and specifically deep neural networks by preventing overfitting to the training dataset. Data overfitting is a common problem when training deep neural networks since they are highly over-parameterized models, which are usually trained using a small amount of data compared to the number of parameters in the model. Regularization helps to reduce the model's complexity and thereby achieve a lower generalization error. For this reason, using regularization techniques is a common practice in the training of machine learning models.

In [6] some insight is given into the role of explicit and implicit regularization in reducing a network's generalization error. Different explicit regularization methods (such as data augmentation, weight decay and dropout) are empirically compared and the conclusion that follows is that explicit regularization is neither a sufficient nor a necessary technique to control the generalization error of a deep neural network. Namely, not using regularization during training does not necessarily mean a larger generalization error will be obtained.

As previously mentioned in Sect. 2, the Rademacher complexity is a complexity measure of a hypothesis set $\mathcal{H}$. It is shown that adding explicit regularization to the training phase (e.g., dropout, weight decay) effectively reduces the Rademacher complexity of the hypothesis space of the possible solutions by confining it to a subspace of the original hypothesis space which has lower complexity. However, this does not necessarily imply a better generalization error, as in most cases, the Rademacher complexity measure is not powerful enough to capture the abilities of deep neural networks.

Similarly, implicit regularization techniques, such as the usage of the SGD algorithm for training, early stopping and batch normalization [16], may also play an important role in improving the generalization capabilities of deep neural networks. Yet, empirical findings show that they are not indispensable for obtaining good generalization.

Deep neural networks can achieve a zero training error even when trained on a random labeling of the training data, meaning deep neural networks can easily fit random labels, which is indicative of very high model capacity. Expanding the scope of this premise, the relation to Convolutional Neural Networks (CNNs) is made by showing that state-of-the-art CNNs for image classification can easily fit a random labeling of the training data, giving further support to the notion that deep neural networks are powerful enough to *memorize* the training data, since randomly labeled data does not encapsulate any actual underlying rule.

These empirical findings are explained using a theoretical result, which shows that a two-layer neural network already has perfect expressivity when the number of parameters exceeds the number of data samples. Specifically, there exists a two-layer neural network with ReLU activations and $2N + D$ weights that can represent any function on a sample of size N in D dimensions.

It follows that training remains a relatively easy task, even for random labels and for data that has been subject to different kinds of random shuffling and permutations, i.e., training is easy even when the model does not generalize well. In addition, it has been empirically established that training on random labels only increases the training time by a small constant factor.

In this context it is important to note that extensive research efforts are still aimed at classical complexity measures such as the Rademacher complexity. For example, [17] provides a bound for the Rademacher complexity of DNNs assuming norm constraints on their weight matrices. Under several assumptions this bound is independent of the network size (width and depth). According to this bound, the Rademacher complexity is upper bounded by[1]

$$\tilde{\mathcal{O}}\left(R\sqrt{\frac{\log\{\frac{R}{\Gamma}\}}{\sqrt{N}}}\right),$$

where R is an upper bound on the product of the Frobenius norms of the network's weight matrices and Γ is a lower bound on the product of the spectral norms of the network's weight matrices.

The work in [6] generally emphasizes the need for a different approach in examining the generalization of DNNs. One specific incentive is the refuted common notion that the widely used regularization techniques are a necessary condition for obtaining good generalization. Moreover, even though several measures have been shown to be correlated with the generalization error of DNNs (as shown in [18] and discussed in Sect. 4.2), there is still a need for tighter bounds and more explanations for the generalization capabilities of DNNs. Therefore, a more comprehensive theory is necessary to explain why DNNs generalize well even though they are capable of memorizing the training data.

4.2 Exploring Generalization in Deep Learning

In [18], several different measures and explanations for the generalization capabilities of DNNs are examined. The examined measures include norm-based control, robustness and sharpness, for which a connection to the PAC-Bayes theory is drawn. The different measures are evaluated based on their theoretical ability to guarantee generalization and their performance when empirically tested.

The capacity of a model for several given metrics (e.g., ℓ_2 distance), which is examined throughout the work in [18], represents the number of training examples necessary to ensure generalization, meaning that the test error is close to the training error.

With similarity to commonly established notions on the matter, it is claimed that using a VC-dimension measure to provide a bound on the capacity of neural networks is insufficient to explain their generalization abilities. Relying on the works in [19, 20], a bound is proposed on the VC-dimension of feedforward neural networks with ReLU activations in terms of the number of parameters in the network. This bound is given by

$$\text{VCdim} = \tilde{\mathcal{O}}(L \cdot \dim(W)),$$

where $\dim(W)$ is the number of parameters in the network and L is the amount of layers in the network. This bound is very loose and therefore fails to explain

[1] $\tilde{\mathcal{O}}$ is the upper bound to the complexity up to a logarithmic factor of the same term.

Table 2 Norm-based capacity measures for the analysis of the generalization of DNNs

Capacity type	Capacity order
ℓ_2-norm	$\frac{1}{\gamma_{margin}^2} \prod_{l=1}^{L} 4\|\|W_l\|\|_F^2$
ℓ_1 path-norm	$\frac{1}{\gamma_{margin}^2} \left(\sum_{j \in \prod_{k=0}^{L}[h_k]} \left\|\prod_{l=1}^{L} 2W_l[j_l, j_{l-1}]\right\|\right)^2$
ℓ_2 path-norm	$\frac{1}{\gamma_{margin}^2} \sum_{j \in \prod_{k=0}^{L}[h_k]} \prod_{l=1}^{L} 4h_l W_l^2[j_l, j_{l-1}]$
Spectral-norm	$\frac{1}{\gamma_{margin}^2} \prod_{l=1}^{L} h_l \|\|W_l\|\|_2^2$

the generalization behavior of these networks. Neural networks are highly overparameterized models, but they can fit a desired function almost perfectly with a training set much smaller than the amount of parameters in the model. For this reason, a bound which is linear in the amount of parameters in the model is too weak to explain the generalization behavior of deep neural networks. We refer the reader to [21] for some earlier work on the VC-dimension of neural networks.

Norm-based complexity measures for neural networks with ReLU activations are presented as well. These measures do not explicitly depend on the amount of parameters in the model and therefore have a better potential to represent its capacity.

Relying on the work in [22], four different norm-based measures are used to evaluate the capacity of deep neural networks. These measures use the following lenient version of the definition of a classification margin: γ_{margin} is the lowest value of γ such that $\lceil \epsilon N \rceil$ data samples have a margin lower than γ, where $\epsilon > 0$ is some small value and N is the size of the training set. The four measures are given in Table 2.

$[h_k]$ is the set $\{h_1, ..., h_k\}$ and $\prod_{k=0}^{L}[h_k]$ is the Cartesian product over the sets $[h_k]$. We remind the reader that h_k is the amount of neurons in the kth layer of the network, and accordingly $h_0 = D$ where D is the network's input dimension and $h_L = K$ where K is the network's output dimension. Note that the ℓ_1 and ℓ_2 path-norms (see Table 2) sum over all possible paths going from the input to the output of the network, passing through one single neuron in each layer. Accordingly, the index j represents a certain path consisting of one neuron in each layer, j_l denotes the neuron in the lth layer in the jth path and $W_l[j_l, j_{l-1}]$ is the weight parameter (scalar) relating the j_{l-1} and the j_l neurons.

The results of empirical tests are presented in [18]. These show that the training error is zero and all the aforementioned norm measures are larger for networks trained to fit random labels than for networks trained to fit the true labels of the data (specifically, the VGG network is used along with the CIFAR-10 dataset). These findings indicate that these norm-based measures can explain the generalization of DNNs, as the complexity of models trained on random labels is always higher than the complexity of models trained on the true labels, corresponding to the favorable generalization abilities of the latter.

Another capacity measure, which is examined is the Lipschitz constant of a network. The question whether controlling the Lipschitz constant of a network with respect to its input can lead to controlling its capacity is investigated, relying on the relation between the weights of a network and its Lipschitz constant. Different bounds on models' complexity based on their Lipschitz constants are reviewed. The relation between a network's Lipschitz constant and the norm of its weights is made, and it is shown that all the bounds relying on the Lipschitz constant result in very loose capacity bounds which are exponential in both the input dimension and the depth of the network. The conclusion is that simply bounding the Lipschitz constant of a network is not enough to obtain a reasonable capacity control for a neural network, and that these bounds are merely a direct consequence of bounding the norm of the network's weights.

In addition, another capacity measure which has been linked to the generalization abilities of DNNs is addressed: the sharpness of the obtained minimizer to the optimization problem of the training of DNNs. It is claimed that the notion of sharpness, which was formulated in [14] (see Definition 4), cannot by itself capture the generalization behavior of neural networks. Instead, a related notion of *expected sharpness* in the context of the PAC-Bayes theorem, combined with the norm of the network's weights, does yield a capacity control that offers good explanations to the observed phenomena.

In a recent related paper [23], a new complexity measure based on unit-wise capacities is proposed. A unit's capacity is defined by the ℓ_2 norm of the difference between the input weights associated with this unit and the initialization values of these weights. These input weights are the multiplicative weights which were applied to the values of the units of the previous layer in order to obtain the value of this unit (i.e., the row that corresponds to this specific unit in the weight matrix of this unit's layer). This complexity measure results in a tighter generalization bound for two-layer ReLU networks.

In another recent work [24] a bound for the statistical error (test error) is given for networks with ℓ_1 constraints on their weights and with ReLU activations. In this setting, the test error is shown to be upper bounded by $\sqrt{\frac{L^3 \log D}{N}}$, where in this context, D is the maximal input dimension to a layer in the network. With this bound, the input dimension can be much larger than the amount of samples, and the learned algorithm would still be accurate if the target function has the appropriate ℓ_1 constraints and N is large compared to $L^3 \log d$.

Following the above, it can be concluded that even though various measures of the capacity of DNNs exist in the literature, some exhibiting good correlation with the generalization abilities of DNNs, a comprehensive theoretical formulation with adequate empirical results that explain the generalization abilities of DNNs is still an active field of research.

4.3 A PAC-Bayesian Approach to Spectrally Normalized Margin Bounds for Neural Networks

In a more recent work [25] a generalization bound for neural networks with ReLU activations is presented in terms of the product of the spectral norm and the Frobenius norm of their weights. A bound on the changes in a network's output with respect to a perturbation in its weights is used to derive a generalization bound. The perturbation bound relies on the following constraint on the input domain:

$$\mathcal{X}_{B,D} = \left\{ x \in \mathbb{R}^D | \sum_{i=1}^{D} x_i^2 \leq B^2 \right\},$$

and is formulated in the following lemma.

Lemma 1 (Perturbation Bound) *For any $B, L > 0$, let $f_W : \mathcal{X}_{B,D} \rightarrow \mathbb{R}^K$ be an L-layer neural network with ReLU activations. Then for any $W = vec(\{W_l\}_{l=1}^L)$, and $x \in \mathcal{X}_{B,D}$, and any perturbation $U = vec(\{U_l\}_{l=1}^L)$ such that $||U_l||_2 \leq \frac{1}{L}||W_l||_2$, the change in the output of the network can be bounded as follows:*

$$||f_{W+U}(x) - f_W(x)||_2 \leq eB \left(\prod_{l=1}^{L} ||W_l||_2 \right) \sum_{l=1}^{L} \frac{||U_l||_2}{||W_l||_2}.$$

This bound makes a direct relation between a perturbation in the model parameters and its effect on the network's output. Therefore, it leads to an upper bound on the allowed perturbation in the model parameters for a desired margin γ. The consequent generalization bound for neural networks with ReLU activations is given in the following theorem.

Theorem 2 *For any $B, L, \alpha > 0$, let $f_W : \mathcal{X}_{B,D} \rightarrow \mathbb{R}^K$ be an L-layer feedforward network with ReLU activations. Then, for any constant δ, margin $\gamma > 0$, network parameters $W = vec(\{W_l\}_{l=1}^L)$ and training set of size N, we have with probability exceeding $1 - \delta$ that*

$$\ell_{exp,0}(f_W) \leq \ell_{emp,\gamma}(f_W) + \mathcal{O}\left(\sqrt{\frac{B^2 L^2 \alpha \ln(L\alpha) \prod_{l=1}^{L} ||W_l||_2^2 \sum_{l=1}^{L} \frac{||W_l||_F^2}{||W_l||_2^2} + \ln \frac{LN}{\delta}}{\gamma^2 N}} \right), \tag{6}$$

where α is an upper bound on the number of units in each layer in the network.

The proof of this theorem relies on the PAC-Bayes theory. We refer the reader to [25] for the full proofs of both Lemma 1 and Theorem 2.

As would have been expected, the bound in (6) becomes tighter as the margin γ increases. In addition, the bound is looser as the number of layers in the network L

increases and for a domain with a larger parameter B, as it encapsulates a larger input domain. With similarity to other generalization error bounds, this bound becomes tighter as the size of the training set increases.

This bound takes a step further toward a comprehensive explanation to the generalization capabilities of DNNs, as unlike many other bounds it incorporates several different measures that define a DNN: the number of layers in the network, the spectral norm of the layers' weights, the number of neurons in each layer, the size of the input domain, the required classification margin and the size of the training set.

Another prominent work [26] provides a margin-based bound for the generalization error of DNNs which scales with their spectral complexity, i.e., their Lipschitz constant (the product of the spectral norms of the weight matrices) times a correction factor. This bound relies on the following definition for the spectral complexity of a network.

Definition 6 (*Spectral Complexity*) A DNN with reference matrices $(M_1, \dots, M_L)$ with the same dimensions as the weight matrices $W_1, \dots, W_L$, for which the nonlinearities ϕ_i are ρ_i-Lipschitz, respectively, has spectral complexity

$$R_W = \left(\prod_{i=1}^{L} \rho_i ||W_i||_2\right)\left(\sum_{i=1}^{L} \frac{||W_i^T - M_i^T||_{2,1}^{\frac{2}{3}}}{||W_i||_2^{\frac{2}{3}}}\right)^{\frac{3}{2}},$$

where here $||\cdot||_2$ represents the spectral norm and the (p, q)-matrix norm is defined by $||\cdot||_{p,q} = ||(||A_{:,1}||_p, \dots, ||A_{:,m}||_p)||_q$ for some matrix $A \in \mathbb{R}^{d\times m}$.

Note that the spectral complexity depends on the chosen reference matrices. This leads to the following margin-based generalization bound. For a DNN with training set $\{(x_i, y_i)\}_{i=1}^N$ drawn i.i.d. from some distribution $\mathcal{D}$, with weight matrices $W_1, \dots, W_L$, for every margin $\gamma > 0$, with probability at least $1 - \delta$ it holds that

$$\ell_{exp,0}(f) \le \ell_{emp,\gamma}(f, x) + \tilde{O}\left(\frac{||X||_2 R_W}{\gamma N} \log(\max_i h_i) + \sqrt{\frac{\log(1/\delta)}{N}}\right)$$

where $||X||_2 = \sqrt{\sum_i ||x_i||_2^2}$, and $\max_i h_i$ represents the maximal amount of neurons in any layer in the DNN. The proof is left to the paper [26].

The PAC-Bayes framework was also examined in [27], where a PAC-Bayes bound on a network's generalization error is optimized in order to obtain non-vacuous (tight, nontrivial) generalization bounds for deep stochastic neural network classifiers. It is hypothesized that the SGD optimization algorithm obtains good solutions only if these solutions are surrounded by a relatively large volume of additional solutions that are similarly good. This leads to the notion that the PAC-Bayes bound has the potential to provide non-vacuous bounds on the generalization error of deep neural networks when it is used optimize the stochastic classifier. As shown in Sect. 2, the PAC-Bayes theorem bounds the expected loss of a stochastic classifier using the Kullback–Leibler divergence between an a priori probability distribution P and a posterior probability distribution Q, from which a classifier is chosen when the training data is available.

An optimization is performed over the distributions Q, in order to find the distribution that minimizes the PAC-Bayes bound. This is done using a variation of the stochastic gradient descent algorithm and using a multivariate Gaussian posterior Q on the network parameters. In each step the network's weights and their corresponding variances are updated by a step in the direction of an unbiased estimate of the gradient of an upper bound on the PAC-Bayes bound. Using this approach, a finer bound on the generalization error of neural networks is obtained. We refer the reader to [27] for the exact mathematical formulation. This approach relates to similar notions examined in other works (such as [14, 28]), that make a connection between the sharpness of the solution obtained using the SGD algorithm and its ability to generalize well.

4.4 Stability and Generalization

The *stability* of a learning algorithm is an important characteristic which represents its ability to maintain similar generalization results when a training example is excluded or replaced in the training dataset. In [10], a sensitivity-driven approach is used to derive generalization error bounds. The sensitivity of learning algorithms to changes in the training set, which are caused by sampling randomness and by noise in the sampled measurements, is formally defined and analyzed throughout [10].

A stable learning algorithm is an algorithm which is *not* sensitive to small changes in the training set, i.e., an algorithm for which a small change in its training set results in a small change in its output. As mentioned earlier, such a small change can be the exclusion or replacement of a certain training example.

Statistical tools of concentration inequalities lead to bounds on the generalization error of stable learning algorithms when the generalization error is essentially treated as a random variable whose expected value is zero when it is constrained to be roughly constant.

The following are several useful definitions for the examination of the influence of changes in the training set $\mathbf{s}$ which consists of N samples independently drawn from the distribution $\mathcal{D}$.

- By excluding (removing) the ith sample in the training set the following set is obtained $\mathbf{s}^{\setminus i} = \{s_1, \ldots, s_{i-1}, s_{i+1}, \ldots, s_N\}$.
- By replacing the ith sample in the training set, the following set is obtained $\mathbf{s}^i = \{s_1, \ldots, s_{i-1}, s_i', s_{i+1}, \ldots, s_N\}$, where the new sample s_i' is drawn from the same distribution $\mathcal{D}$ and is independent from $\mathbf{s}$.

The following analysis is based on inequalities that relate moments of multidimensional random functions to their first order finite differences. Let us present the following four definitions for the stability of a learning algorithm. These definitions will be later used to derive generalization bounds.

Definition 7 An algorithm $\mathcal{A}$ has hypothesis stability β with respect to the loss function ℓ if the following holds:

$$\forall i \in \{1, \ldots, N\}, \mathbb{E}_{\mathbf{s},s}\left[|\ell(\mathcal{A}_{\mathbf{s}}, s) - \ell(\mathcal{A}_{\mathbf{s}^{\setminus \mathbf{i}}}, s)|\right] \leq \beta.$$

The hypothesis stability relates to the average change caused by the exclusion of one training sample. In order to limit the change at every specific training point, the following definition of point-wise hypothesis stability is presented.

Definition 8 An algorithm $\mathcal{A}$ has point-wise hypothesis stability β with respect to the loss function ℓ if the following holds:

$$\forall i \in \{1, \ldots, N\}, \mathbb{E}_{\mathbf{s}}[|\ell(\mathcal{A}_{\mathbf{s}}, s_i) - \ell(\mathcal{A}_{\mathbf{s}^{\setminus \mathbf{i}}}, s_i)|] \leq \beta.$$

For measuring the change in the expected error of an algorithm instead of the point-wise change, the following definition of error stability, which satisfies a weaker notion of stability, is presented.

Definition 9 An algorithm $\mathcal{A}$ has error stability β with respect to the loss function ℓ if the following holds:

$$\forall \mathbf{s} \in \mathcal{Z}^N, \forall i \in \{1, \ldots, N\}, |\mathbb{E}_s[\ell(\mathcal{A}_{\mathbf{s}}, s)] - \mathbb{E}_s[\ell(\mathcal{A}_{\mathbf{s}^{\setminus \mathbf{i}}}, s)]| \leq \beta.$$

Lastly, the uniform stability, which is a stronger definition of stability, leads to tight bounds.

Definition 10 An algorithm $\mathcal{A}$ has uniform stability β with respect to the loss function ℓ if the following holds:

$$\forall \mathbf{s} \in \mathcal{Z}^N, \forall i \in \{1, \ldots, N\}, ||\ell(\mathcal{A}_{\mathbf{s}}, s) - \ell(\mathcal{A}_{\mathbf{s}^{\setminus \mathbf{i}}}, s)||_{\infty} \leq \beta.$$

Using these four definitions of the stability of learning algorithms, the following bounds on the relation between the empirical loss and the expected loss are derived. Let us denote the leave-one-out loss on the training set by $\ell_{loo}(\mathcal{A}_{\mathbf{s}}) \triangleq \frac{1}{N}\sum_{i=1}^{N} \ell\left(\mathcal{A}_{\mathbf{s}^{\setminus i}}, s_i\right)$. This loss is of importance when discussing an algorithm's stability since it represents its average test loss on a specific sample when it is excluded from its training set.

The following are polynomial bounds on the expected loss.

Theorem 3 *For any learning algorithm $\mathcal{A}$ with hypothesis stability β_1 and point-wise hypothesis stability β_2 with respect to a loss function ℓ such that $0 \leq \ell(f(x), y) \leq M$, we have with probability $1 - \delta$,*

$$\ell_{exp}(\mathcal{A}_{\mathbf{s}}) \leq \ell_{emp}(\mathcal{A}_{\mathbf{s}}) + \sqrt{\frac{M^2 + 12MN\beta_2}{2N\delta}},$$

and

$$\ell_{exp}(\mathcal{A}_{\mathbf{s}}) \leq \ell_{loo}(\mathcal{A}_{\mathbf{s}}) + \sqrt{\frac{M^2 + 6MN\beta_1}{2N\delta}}.$$

We refer the reader to [10] for the proofs. Specifically, for the regression and classification cases, bounds based on the uniform stability of learning algorithms are derived. The bounds for the latter case are left to the original paper, whereas the bounds for the former case are as follows.

Theorem 4 *Let $\mathcal{A}$ be an algorithm with uniform stability β with respect to a loss function ℓ such that $0 \leq \ell(\mathcal{A}_{\mathbf{s}}, s) \leq M$, for all $s \in \mathcal{Z}$ and all sets $\mathbf{s}$. Then, for any $N \geq 1$, and any $\delta \in (0, 1)$, the following bounds hold (separately) with probability at least $1 - \delta$ over the random draw of the sample $\mathbf{s}$:*

$$\ell_{exp}(\mathcal{A}_{\mathbf{s}}) \leq \ell_{emp}(\mathcal{A}_{\mathbf{s}}) + 2\beta + (4N\beta + M)\sqrt{\frac{\ln(1/\delta)}{2N}},$$

and

$$\ell_{exp}(\mathcal{A}_{\mathbf{s}}) \leq \ell_{loo}(\mathcal{A}_{\mathbf{s}}) + \beta + (4N\beta + M)\sqrt{\frac{\ln(1/\delta)}{2N}}.$$

This theorem gives tight bounds when the stability β scales as $\frac{1}{N}$, which is the case for several prominent algorithms such as the K-NN classifier with respect to the $\{0,1\}$ loss function, and the SVM classifier with respect to the Hinge loss function. Bounds for the case when regularization is used can be controlled by the regularization parameter (a scalar hyperparameter, usually denoted by λ, which controls the weight of the regularization term in the objective loss function) and can therefore be very tight. These bounds are left to the original paper [10].

The specific relation to deep neural networks was made in [29], where several theorems regarding the stability of deep neural networks are given. It is shown that stochastic gradient methods, which are the most commonly used methods for training DNNs, are stable. Specifically, the following theorem establishes that stochastic gradient methods are uniformly stable.

Theorem 5 *Assume that $\ell(x) \in [0, 1]$ is an M-Lipschitz and $\epsilon - smooth$ loss function for every x. Suppose that we run the stochastic gradient method for T steps with monotonically nonincreasing step sizes $\alpha_t \leq \frac{c}{t}$. Then, the stochastic gradient method applied to ℓ has uniform stability with*

$$\beta_{stability} \leq \frac{1 + \frac{1}{\epsilon c}}{N - 1}(2cM^2)^{\frac{1}{\epsilon c+1}} T^{\frac{\epsilon c}{\epsilon c+1}},$$

where a function $\ell(x)$ is M-Lipschitz if for all points x in the domain of ℓ it holds that $||\nabla\ell(x)||_2 \leq M$, and a function $\ell(x)$ is $\epsilon - smooth$ if for all $x, \hat{x}$ in the domain of ℓ it holds that $||\nabla\ell(x) - \nabla\ell(\hat{x})||_2 \leq \epsilon||x - \hat{x}||_2$.

Specifically, if the constant factors that depend on ϵ, c and M are omitted, the bound on the uniform stability is given by

$$\beta_{stability} \lessapprox \frac{1}{N} T^{1-\frac{1}{\epsilon c+1}}.$$

This bound implies that under certain assumptions on the loss function the uniform stability β scales as $\frac{1}{N}$ for deep neural networks, and in this case, the same bounds from Theorem 4 are tight for deep neural networks as well. We refer the reader to [29] for the formal proof of this theorem.

The notion of stability has therefore been shown to be of importance in the evaluation of a learning algorithm's generalization error. It has been established that stable algorithms yield a lower expected loss and therefore a lower generalization error, particularly for deep neural networks which in many cases can obtain a training loss of zero (as shown in [6]). A comprehensive overview of [29] is given in Sect. 4.7.

4.5 Robustness and Generalization

In a later work [11], a notion from robust optimization theory is used to examine the generalization capabilities of learning algorithms with respect to their robustness. An algorithm is said to be robust if it achieves similar performance on a test sample and a training sample which are close in some sense, i.e., if a test sample is similar to a training sample, then its corresponding test error is close to the corresponding training error. This means that a robust learning algorithm is not sensitive to small perturbations in the training data. This notion applies to general learning algorithms, not only deep neural networks. The formal definition of algorithm robustness is given in Sect. 2. The following is the generalization error bound for a robust learning algorithm.

Theorem 6 *If* $\mathbf{s}$ *consists of* N *i.i.d. samples, and* $\mathcal{A}$ *is* $(K, \epsilon(\mathbf{s}))$*-robust, then for any* $\delta > 0$, *with probability at least* $1 - \delta$,

$$|\ell_{exp}(\mathcal{A}_{\mathbf{s}}) - \ell_{emp}(\mathcal{A}_{\mathbf{s}})| \leq \epsilon(\mathbf{s}) + M\sqrt{\frac{2K\ln 2 + 2\ln(1/\delta)}{N}}.$$

This holds under the assumption that $\ell(\mathcal{A}_{\mathbf{s}})$ is nonnegative and upper bounded uniformly by the scalar M.

A new relaxed definition of pseudo-robustness (weak robustness) is proposed. Pseudo-robustness is both a necessary and sufficient condition for asymptotic generalizability of learning algorithms in the limit superior sense (as shown in Definitions 12 and 13 hereafter). Under the definition of pseudo-robustness, the condition for robustness as mentioned in (3) in the preliminaries, only has to hold for a subset of the training samples. The definition of pseudo-robustness is as follows.

Definition 11 Algorithm $\mathcal{A}$ is $(K, \epsilon(\mathbf{s}), \hat{N})$ pseudo-robust if $\mathcal{Z}$ can be partitioned into K disjoint sets, denoted as $\{C_i\}_{i=1}^{K}$, and a subset of training samples $\hat{\mathbf{s}}$ with $|\hat{\mathbf{s}}| = \hat{N}$ such that $\forall s \in \hat{\mathbf{s}}$,

$$s, z \in C_i, \Rightarrow |\ell(\mathcal{A}_{\mathbf{s}}, s) - \ell(\mathcal{A}_{\mathbf{s}}, z)| \leq \epsilon(\mathbf{s}).$$

The following theorem gives a bound on the generalization error of pseudo-robust learning algorithms.

Theorem 7 *If* $\mathbf{s}$ *consists of* N *i.i.d. samples, and* $\mathcal{A}$ *is* $(K, \epsilon(\mathbf{s}), \hat{N})$ *pseudo-robust, then for any* $\delta > 0$*, we have that with a probability at least* $1 - \delta$*,*

$$\left|\ell_{exp}(\mathcal{A}_{\mathbf{s}}) - \ell_{emp}(\mathcal{A}_{\mathbf{s}})\right| \leq \frac{\hat{N}}{N}\epsilon(s) + M\left(\frac{N - \hat{N}}{N} + \sqrt{\frac{2K\ln 2 + 2\ln(1/\delta)}{N}}\right).$$

This holds under the assumption that $\ell(\mathcal{A}_{\mathbf{s}})$ is nonnegative and upper bounded uniformly by a scalar M. The proof is offered in its entirety in [11].

Robustness is an essential property for successful learning. In particular, pseudo-robustness (weak robustness) is indicative of the generalization abilities of a learning algorithm. A learning algorithm generalizes well if and only if it is pseudo-robust. This conclusion is formalized by the following definitions.

Definition 12 1. A learning algorithm $\mathcal{A}$ generalizes w.r.t. $\mathbf{s}$ if

$$\limsup_{N} \left\{\mathbb{E}_t\left(\ell(\mathcal{A}_{\mathbf{s}_N}, t)\right) - \frac{1}{N}\sum_{i=1}^{N}\ell(\mathcal{A}_{\mathbf{s}_N}, s_i)\right\} \leq 0.$$

2. A learning algorithm $\mathcal{A}$ generalizes with probability 1 if it generalizes w.r.t. almost every $\mathbf{s}$.

Definition 13 1. A learning algorithm $\mathcal{A}$ is weakly robust w.r.t. $\mathbf{s}$ if there exists a sequence of $\{\mathcal{D}_N \subseteq \mathcal{Z}^N\}$ such that $Pr\left(\mathbf{t}_N \in \mathcal{D}_N\right) \rightarrow 1$, and

$$\limsup_{N} \left\{\max_{\hat{\mathbf{s}}_N \in \mathcal{D}_N}\left[\frac{1}{N}\sum_{i=1}^{N}\ell(\mathcal{A}_{\mathbf{s}_N}, \hat{s}_i) - \frac{1}{N}\sum_{i=1}^{N}\ell(\mathcal{A}_{\mathbf{s}_N}, s_i)\right]\right\} \leq 0. \tag{7}$$

2. A learning algorithm $\mathcal{A}$ is asymptotically weakly robust if it is robust w.r.t. almost every $\mathbf{s}$.

Note that $\mathcal{A}_{\mathbf{s}_N}$ is the learning algorithm $\mathcal{A}$ trained on the set $\mathbf{s}_N = \{s_1, \ldots, s_N\}$, and $\hat{\mathbf{s}}_N = \{\hat{s}_1, \ldots, \hat{s}_N\} \in \mathcal{D}_N$ is the sequence of samples. It follows from (7) that if for a large subset of $\mathcal{Z}^N$ the test error is close to the training error, then the learning algorithm is pseudo-robust (weakly robust). The thorough proof is offered in [11].

The following theorem is given to make a general relation between pseudo-robustness of a learning algorithm and its generalization capabilities.

Theorem 8 *An algorithm $\mathcal{A}$ generalizes w.r.t.* **s** *if and only if it is weakly robust w.r.t.* **s**.

The following corollary stems from the aforementioned theorem and further formalizes the discussed relation.

Corollary 1 *An algorithm $\mathcal{A}$ generalizes with probability 1 if and only if it is asymptotically weakly robust.*

Therefore, it has been established that weak robustness is a fundamental characteristic for learning algorithms to be able to generalize well.

In order to make a relation between the above theorems and feedforward neural networks, we introduce the *covering number* term as defined in [30].

Definition 14 (*Covering number*) For a metric space $\mathcal{S}$ with metric d and $\mathcal{X} \subset \mathcal{S}$, it is said that $\hat{\mathcal{X}} \subset \mathcal{S}$ is a ρ-cover of $\mathcal{X}$ if $\forall x \in \mathcal{X}, \exists \hat{x} \in \hat{\mathcal{X}}$ such that $d(x, \hat{x}) \leq \rho$. The ρ-covering number of the space $\mathcal{X}$ with d-metric balls of radius ρ is

$$\mathcal{N}(\rho, \mathcal{X}, d) = \min\{|\hat{\mathcal{X}}| \text{s.t.}\ \hat{\mathcal{X}} \text{ is a } \rho\text{−cover of } \mathcal{X}\}.$$

Accordingly, the term $\mathcal{N}(\frac{\gamma}{2}, \mathcal{Z}, || \cdot ||_\infty)$, which is used in the following example, represents the $\frac{\gamma}{2}$-covering number of the space $\mathcal{Z}$ with the metric $|| \cdot ||_\infty$.

The following example makes the relation to deep neural networks.

Example 3 Let $\mathcal{Z}$ be compact and the loss function on the sample $s = (x, y)$ be $\ell(\mathcal{A}_\mathbf{s}, s) = |y - \mathcal{A}_\mathbf{s}(x)|$. Consider the L-layer neural network trained on **s**, which is the following predicting rule given an input $x \in \mathcal{X}$

$$\forall l = 1, \ldots, L-1 : x_i^l \triangleq \phi\left(\sum_{j=1}^{h_{l-1}} w_{ij}^{l-1} x_j^{l-1}\right); i = 1, \ldots, h_l; \tag{8}$$

$$\mathcal{A}_\mathbf{s}(x) \triangleq \phi\left(\sum_{j=1}^{h_{L-1}} w_j^{L-1} x_j^{L-1}\right). \tag{9}$$

If there exist α, β such that the L-layer neural network satisfies that $|\phi(a) - \phi(b)| \leq \beta|a - b|$, and $\sum_{j=1}^{h_l} |w_{ij}^l| \leq \alpha$ for all l, i, then it is $\left(\mathcal{N}(\frac{\gamma}{2}, \mathcal{Z}, || \cdot ||_\infty), \alpha^L \beta^L \gamma\right)$-robust, for all $\gamma > 0$.

Note that $x^0 \triangleq x$ represents the network's input, and Eqs. (8), (9) depict standard data propagation through the network. An interesting result is that the number of neurons in each layer does not affect the robustness of the algorithm, and as a result the test error.

In [31], the notion of robustness from [11] is used to derive a bound on the generalization error of DNN classifiers trained with the 0–1 loss, where the sample space $\mathcal{X}$ is a subset of a C_M regular D-dimensional manifold whose covering number is upper bounded by $\mathcal{N}(\rho, \mathcal{X}, d) \leq (\frac{C_M}{\rho})^D$. In this case, the advantage of the robustness framework is that it provides a connection between the generalization error of the classifier and the data model. Yet, the bound provided in [31] scales exponentially with the intrinsic dimension of the data. Therefore, it is a rather loose bound and a tighter bound is required to better explain the generalization capabilities of DNN classifiers.

4.6 Stronger Generalization Bounds for Deep Nets via a Compression Approach

A compression-based approach has recently been proposed to derive tight generalization bounds for deep neural networks [15]. This proposed compression is essentially a re-parameterization of the trained neural network, which relies on compression algorithms for reducing the effective number of parameters in deep neural networks. Using noise stability properties a theoretical analysis of this compression-based approach leads to tight generalization bounds. Generalization bounds that apply to Convolutional Neural Networks (CNNs) are also drawn for these compressed networks, and the correlation to their generalization capabilities is empirically established.

First, it is shown in [15] that Gaussian noise injected to different layers in a neural network has a rapidly decaying impact on the following layers. This attenuation of the noise as it propagates through the network layers implies a noise stability that allows the compression of individual layers of the network. The definition of a compressible classifier is given in Sect. 2.

Incorporating the use of a “helper string” s, which is essentially a vector of fixed arbitrary numbers, enables the compression of the difference between the final weights and the helper string, instead of the weights themselves. The usage of a helper string yields tighter generalization bounds, such as in [27].

Definition 15 Let $G_{\mathcal{W},s} = \{g_{W,s} | W \in \mathcal{W}\}$ be a class of classifiers with trainable parameters W and a helper string s. A classifier f is $(\gamma, \mathbf{s})$-compressible with respect to $G_{\mathcal{W},s}$ if there exists $W \in \mathcal{W}$ such that for any sample in the dataset $x \in \mathbf{s}$, we have for all k

$$|f(x)[k] - g_{W,s}(x)[k]| \leq \gamma.$$

Note that $f(x)[k]$ is the kth entry in the K-dimensional vector $f(x)$. The aforementioned definition leads to the following theorem for a general classifier.

Theorem 9 *Suppose $G_{\mathcal{W},s} = \{g_{W,s} | W \in \mathcal{W}\}$, where W is a set of q parameters, each of which can have at most r discrete values, and s is a helper string. Let* $\mathbf{s}$ *be a training set with N samples. If the trained classifier f is* $(\gamma, \mathbf{s})$*-compressible with $G_{\mathcal{W},s}$, then there exists $W \in \mathcal{W}$ for which with high probability over the training set,*

$$\ell_{exp,0}(g_{W,s}) \leq \ell_{emp,\gamma}(f, \mathbf{s}) + \mathcal{O}\left(\sqrt{\frac{q \log r}{N}}\right).$$

This theorem formalizes the generalization abilities of the compression of a classifier f. Relying on this finding, a compression algorithm which yields a bounded generalization error on the output of deep neural networks is proposed in [15]. This compression algorithm changes the weights of the neural network using a variation of matrix projection. We refer the reader to [15] for the corresponding bound for CNNs, along with the empirical findings that establish that it is tighter than the familiar ones based on the product of the weight matrix norms, which are shown to be quite loose.

Another recent work [32] also takes a compression-based approach to examining the generalization of DNNs, and provides some interesting complementary insights. A generalization bound for compressed networks based on their compressed size is given, and it is shown that the compressibility of models that tend to overfit is limited, meaning more bits would be necessary to save a trained network which overfits its training dataset.

4.7 Train Faster, Generalize Better: Stability of Stochastic Gradient Descent

The approach of examining generalization through the lens of the commonly used stochastic gradient optimization methods is taken in [29]. It is essentially claimed that any model trained with a stochastic gradient method for a reasonable amount of time would exhibit a small generalization error.

Much insight is given into why the usage of stochastic gradient methods yields good generalization in practice, along with a formal foundation as to why popular techniques for training deep neural networks promote the stability of the obtained solution. It is argued that stochastic gradient methods are useful in achieving a low generalization error since as long as the objective function is smooth and the number of taken steps is sufficiently small these methods are stable. Relying on the definitions and bounds for algorithm stability from [10], stability bounds for both convex and non-convex optimization problems are derived under standard Lipschitz and smoothness assumptions.

An interesting aspect is the relation between an algorithm's generalization error and the amount of training epochs used during its optimization process. When an algorithm is trained for an arbitrarily long training time, it could achieve a small training error by memorizing the training dataset, yet with no generalization abilities.

However, an algorithm's ability to fit the training data *rapidly*, with a reasonably small amount of training iterations, is indicative of its ability to generalize well.

It is shown that stochastic gradient methods are uniformly stable. In the convex case, the stability measure decreases as a function of the sum of the optimization step sizes, meaning that these methods reach a solution that generalizes well as long as the optimization steps are sufficiently small and the number of iterations is not too large. Moreover, for strictly convex loss functions, these methods are stable for an arbitrarily long training time. Relating to the non-convex case of neural networks, it is shown that the number of training steps of stochastic gradient methods can grow as fast as N^c for N training samples and a small $c > 1$, and good generalization would still be achieved. This sheds light on the superior generalization abilities of neural networks, which are trained with many optimization steps.

The following theorem gives a bound for convex loss minimization with a stochastic gradient method.

Theorem 10 *Let the loss function ℓ be ϵ-smooth, convex and M-Lipschitz. Then a stochastic gradient method with step sizes $\alpha_t \leq \frac{2}{\epsilon}$ for T steps satisfies uniform stability with*

$$\beta_{stability} \leq \frac{2M^2}{N} \sum_{t=1}^{T} \alpha_t.$$

We leave the formal proof of this theorem to [29]. We refer the reader to the corresponding stability bound for the non-convex case given in Sect. 4.4.

Moreover, as long as the number of training iterations is linear in the number of data points in the training set, the generalization error is bounded by a vanishing function of the sample size. This means that a short training time by itself can be sufficient to prevent overfitting, even for models with a large amount of trainable parameters and no explicit regularization.

In addition, a theoretical affirmation is given to the familiar role of regularization in reducing overfitting and improving the generalization capabilities of learning algorithms. The advantages of using methods such as weight decay regularization, gradient clipping, dropout, projection, etc., are formulated and explained.

For instance, the popular technique of dropout decreases the effective Lipschitz constant of the objective function, thus decreasing the bound on the generalization error, as formalized in the following theorem.

Theorem 11 *A randomized map $D : \Omega \to \Omega$ is a dropout operator with rate r if for every $v \in D$ it holds that $\mathbb{E}\{||Dv||_2\} = r||v||_2$. For a differentiable function $f : \Omega \to \Omega$, which is M-Lipschitz, the dropout gradient update defined by $\alpha D(\nabla f(v))$, with learning rate α is $(r\alpha M)$-bounded.*

Proof Since f is assumed to be differentiable and M-Lipschitz, using the linearity of the expectation operator we get that

$$\mathbb{E}\{||\alpha D(\nabla f(v))||\} = \alpha r \mathbb{E}||\nabla f(v)|| \leq \alpha r M.$$

This obtained upper bound to the gradient update implies an enhanced stability of the learning algorithm according to the dependency on the Lipschitz constant M, which appears in various generalization bounds.

Many other works analyze the characteristics of the loss function and the SGD optimization algorithm used for the training of DNNs as well. A recent work [33] shows that even without explicit regularization, for linearly separable logistic regression problems the SGD algorithm converges to the same direction as the max-margin solution, i.e., the solution of the hard margin SVM.

Another recent work [34] studies the problem of two-layer neural networks with ReLU or Leaky ReLU activations when the data is linearly separable. In the specific examined setting, the parameters of the first layer are updated whereas the parameters of the second layer are fixed throughout the training process. Convergence rates of the SGD algorithm to a global minimum are introduced and generalization guarantees for this minimum, which are independent of the network size, are given as well.

Another related work [35] examines why DNN architectures that have multiple branches (e.g., Inception, SqueezeNet, Wide ResNet and more) exhibit improved performance in many applications. It is claimed that one cause for this phenomenon is the fact that multi-branch architectures are less non-convex in terms of the duality gap of the optimization problem in comparison to other commonly used DNN architectures. This may explain why the usage of stochastic gradient methods yields improved generalization results for these networks, as it may contribute to their improved stability.

4.8 On Large Batch Training for Deep Learning: Generalization Gap and Sharp Minima

In [14], another point of view on stochastic gradient methods is taken through the examination of the effect of the size of the training mini-batch on the generalization capabilities of the obtained solution. Though this point of view is mostly empirical, it offers thought-provoking explanations to an interesting phenomenon.

SGD based algorithms perform an optimization step using the gradient of the objective function which is computed on a subset of the training dataset, commonly referred to as a training “mini-batch”. In deep learning, typical mini-batch sizes for training are between several tens to several hundreds of training samples per mini-batch. It has been empirically observed that training using a larger mini-batch, i.e., more training samples are used to make an optimization step in each iteration, leads to a larger generalization error of the obtained solution.

In [14], an explanation to this phenomenon is given by the notion that the usage of large mini-batches encourages convergence to sharp minima solutions to the optimization problem of the training of DNNs (i.e., minimizers of the training loss function), thus obtaining worse generalization. Contrastingly, the usage of small mini-batches tends to lead to solutions with flat minima which yield better generalization. For the exact definition of the term *sharpness* in this context, see Definition 4.

Much light is shed on the aforementioned phenomenon by examining the sharpness of the obtained solutions (minimizers). It is empirically shown that using large mini-batches during training leads to convergence to a solution with large sharpness, whereas training with small mini-batches leads to a solution with small sharpness (large flatness), which has been linked to better generalization. The value of the minimum itself (the value of the objective function at the minimizer) in both cases is often very similar, despite the difference in sharpness.

One explanation is that using smaller mini-batches for gradient based steps is essentially equivalent to using a noisy approximation of the gradient, a property that generally leads to convergence to a flatter minimum. Other conjectures claim that large mini-batches encourage overfitting, are attracted to saddle points, or simply lack the "explorative" characteristic of small mini-batches.

It is shown in [14] that the usage of large mini-batches leads to convergence to sharp minimizers of the objective function, i.e., the Hessian matrix $\nabla^2 \ell_{emp}(\mathcal{A}_{\mathbf{s}_N})$ has a significant amount of large positive eigenvalues, whereas small mini-batches lead to convergence to flat minimizers, with many small eigenvalues in $\nabla^2 \ell_{emp}(\mathcal{A}_{\mathbf{s}_N})$. This leads to the conclusion that large mini-batches effectively prevent the stochastic gradient optimization method from evading undesired basins of attractions.

For the empirical analysis of this phenomenon, an analysis of the Hessian matrix $\nabla^2 \ell_{emp}(\mathcal{A}_{\mathbf{s}_N})$ is required. Due to the significant computational overhead of computing the Hessian matrix in deep learning models, an alternative sharpness measure is employed. We again refer the reader to Definition 4 in the preliminaries for the specifics.

Using this metric of sharpness, it is shown on six different networks that there is a strong correlation between the usage of small training mini-batches and flatness (i.e., low sharpness) of the obtained solution which leads to a small generalization error (empirically evaluated by the difference between the test and training errors). Returning to the notion of the usage of small mini-batches being equivalent to noisy approximation of the gradient of the loss function, it is presumed that during training this noise effectively encourages the objective function to exit basins of attraction of sharp solutions toward the ones belonging to flat solutions. Contrastingly, large mini-batches do not have sufficient noise in the gradient to escape sharp minimizers, thus leading to convergence to a worse solution in the sense of a larger generalization error.

Training using a larger mini-batch is highly beneficial, as it allows an increased parallelization of the performed computations, and faster training as a result. For this reason, several mitigation methods for this phenomenon, that will allow the usage of larger mini-batches during training without compromising the generalization capabilities of the obtained solution, are presented. These methods improve the performance of solutions which are obtained using large mini-batch training. Such methods are data augmentation, conservative training, robust training and others. However, these methods exhibit a limited influence on the sharpness of the attained solution and thereby a limited influence on the generalization error.

4.9 Sharp Minima Solutions to the Training of DNNs Can Generalize for Deep Nets

In [28], the notion that the flatness of the minima of the loss function obtained using SGD-based optimization algorithms is key in achieving good generalization is examined. The relation between the geometry of the loss function in the environment of a solution and the obtained generalization error is examined, and through the exploration of different definitions of "flatness" substantial insight is provided into the conjecture that flat minima lead to better generalization.

With contrast to other prominent works, it is argued that most notions of flatness cannot be directly used to explain generalization. It is specifically shown that for DNNs with ReLU activations it is possible to apply model re-parameterization and obtain arbitrarily sharper minima. This essentially means that the notion of flatness can be abstract, and different interpretations of it could lead to very different conclusions. The reason for this contradiction stems from Bayesian arguments regarding the KL divergence, which are used to explain the superior generalization ability of flat minima. Since the KL divergence is invariant to parameter change, and the notion of flatness is not characterized by such invariance, arguments of flatness can be mistakenly made when more context regarding the definition of flatness is absent.

In this aspect, the work in [28] nicely exhibits that even though empirical evidence points to a correlation between flat minima and good generalization, the exact definition of "flatness" in this context is important, as different definitions can lead to very different results and subsequent conclusions.

Several related properties of the Hessian of deep neural networks, their generalization capabilities and the role of the SGD-based optimization are also examined in [36].

4.10 Train Longer, Generalize Better: Closing the Generalization Gap in Large Batch Training of Neural Networks

In an attempt to tackle the phenomenon of degraded generalization error when large mini-batch training is used, a different theoretical explanation, along with a consequent technique to overcome the phenomenon, is suggested in [37]. It is shown that by adjusting the learning rate and using batch normalization during training, the generalization gap between small and large mini-batches can be significantly decreased. In addition, it is claimed that there is no actual generalization gap between these two cases; large mini-batch training can generalize just as well as small mini-batch training by adapting the number of training iterations to the mini-batch size.

This claim relies on the conjecture that the initial training phase of a neural network using a stochastic gradient method can be described as a high-dimensional "random walk on a random potential" process with "ultra-slow" logarithmic increase in the distance of the weights from their initialization values. Empirical results show that small mini-batch training produces network weights that are further away from their initial values, compared to the case of large mini-batch training. Consequently, by adjusting the learning rate and adding batch normalization to the training algorithm, the generalization gap between small and large mini-batch training can be substantially decreased. This also implies that the initial training phase with a high learning rate is crucial in obtaining good generalization. Training longer in the initial high learning rate regime enables the model to reach farther environments in the objective function space, which may explain why it allows the optimization algorithm to find a flatter minima which is correlated with better generalization.

This leads to the conclusion that there is no inherent generalization gap in this case: adapting the amount of training iterations can mitigate the generalization gap between small and large mini-batch training. Based on these findings, the "Ghost Batch Normalization" algorithm for training using large mini-batches is presented.

```
**Algorithm: (Ghost Batch Normalization)** Inputs: activation values $x$ over a large mini-batch $B_{large} = \{x_1, \dots, x_m\}$ of size $|B_{large}|$, size of virtual small mini-batch $|B_{small}|$ (where $|B_{small}| < |B_{large}|$). $\gamma$, $\beta$, $\eta$ are learned algorithm parameters, where $\eta$ represents the learning momentum.

Training Phase:
Scatter $B_{large}$ s.t.

    $\{X^1, X^2, \dots, X^{\frac{|B_{large}|}{|B_{small}|}}\} = \{x_{1,\dots,|B_{small}|}, x_{|B_{small}|+1,\dots,2|B_{small}|}, \dots, x_{|B_{large}|-|B_{small}|,\dots,m}\}$

$\mu_B^l \leftarrow \frac{1}{|B_{small}|} \sum_{i=1}^{|B_{small}|} X_i^l$   for $l = 1, 2, 3, \dots$ (ghost mini-batch means)
$\sigma_B^l \leftarrow \sqrt{\frac{1}{|B_{small}|} \sum_{i=1}^{|B_{small}|} \left(X_i^l - \mu_B^l\right)^2 + \epsilon}$   for $l = 1, 2, 3, \dots$ (ghost mini-batch std's)
$\mu_{run} = (1-\eta)^{|B_{small}|} \cdot \mu_{run} + \sum_{i=1}^{\frac{|B_{large}|}{|B_{small}|}} (1-\eta)^i \cdot \eta \cdot \mu_B^l$
$\sigma_{run} = (1-\eta)^{|B_{small}|} \cdot \sigma_{run} + \sum_{i=1}^{\frac{|B_{large}|}{|B_{small}|}} (1-\eta)^i \cdot \eta \cdot \sigma_B^l$
return $\gamma \cdot \frac{X^l - \mu_B^l}{\sigma_B^l} + \beta$

Test Phase:
return $\gamma \cdot \frac{X^l - \mu_{run}^l}{\sigma_{run}^l} + \beta$   (scale & shift)
```

This algorithm enables a decrease in the generalization error without increasing the overall number of parameter updates as it acquires the necessary statistics on small virtual ("ghost") mini-batches instead of the original larger mini-batches.

In addition, common practice instructs that during training, when the test error plateaus, one should decrease the learning rate or stop training all together to avoid overfitting. However, in [37] it has been empirically observed that continuing to train, even when the training error decreases and the test error stays roughly the same, results in a test error decrease at a later stage of training when the learning rate is decreased, which is indicative of better generalization.

These results provided the incentive to make the relation to the mini-batch size in [37], supporting the idea that the problem is not in the mini-batch size but rather in the number of training updates. By prolonging the training time for larger mini-batch training by a factor of $\frac{|B_{large}|}{|B_{small}|}e$, where $|B_{large}|$ and $|B_{small}|$ are the sizes of the large and small training mini-batches respectively and e is the amount of training epochs in the original regime, it is empirically shown how the generalization gap between the two cases can be completely closed.

4.11 Generalization Error of Invariant Classifiers

In [38], the generalization error of invariant classifiers is studied. A common case in the field of computer vision is the one in which the classification task is invariant to certain transformations of the input such as viewpoint, illumination variation, rotation, etc. The definition of an invariant algorithm is as follows.

Definition 16 (*Invariant Algorithm*) A learning algorithm $\mathcal{A}$ is invariant to the set of transformations $\mathcal{T}$ if its embedding is invariant:

$$f(t_i(x), \mathbf{s}_N) = f(t_j(x), \mathbf{s}_N) \;\; \forall x \in \mathcal{X}, t_i, t_j \in \mathcal{T}.$$

where $\mathcal{X}$ is the algorithm's input space and t_i, t_j are transformations.

It is shown that invariant classifiers may have a much smaller generalization error than non-invariant classifiers, and a relation to the size of the set of transformations that a learning algorithm is invariant to is made. Namely, it is shown that given a learning method invariant to a set of transformations of size T, the generalization error of this method may be up to a factor $\sqrt{T}$ smaller than the generalization error of a non-invariant learning method. We leave the details of the proof to [38].

Many other works examine invariant classifiers, as utilizing the property of invariance can lead to improved algorithm performance. Another prominent work that examines invariant image representations for the purpose of classification is [39].

4.12 Generalization Error and Adversarial Attacks

It has lately been shown that even though deep neural networks typically obtain a low generalization error, and therefore perform their designated tasks with high accuracy, they are highly susceptible to adversarial attacks [40, 41], with similarity to other machine learning algorithms. An adversarial attack is a perturbation in the model's input which results in its failure. Adversarial attacks have been shown to be very effective: even when the change in the input is very small they are likely to fool the model, and are usually unnoticeable to the human eye. On top of that, very little knowledge of the attacked network is necessary for an efficient attack to be crafted, and once an adversarial example is obtained it is highly transferable, meaning it is very likely to fool other DNNs as well.

The existence of adversarial attacks exposes an inherent fault in DNN models and their ability to generalize well: although DNNs can generalize very well, they can be very easily fooled. One should keep in mind that this fault is not unique to deep neural networks and characterizes other machine learning models as well.

In [31], a new regularization technique is suggested using the regularization of the Frobenius norm of a network's Jacobian matrix. It is shown that bounding the Frobenius norm of the network's Jacobian matrix reduces the obtained generalization error. In [42], it is shown that neural networks are more robust to input perturbations in the vicinity of the training data manifold, as measured by the norm of the network's Jacobian matrix. The correlation between the aforementioned robustness and the network's generalization capabilities is also noted. In [43], this notion is taken further. It is shown that this Jacobian regularization also improves the robustness of DNNs to adversarial attacks, thus showing that reducing a network's generalization error has also collateral benefits. In [44], it is shown that the sample complexity (the number of training samples necessary to learn the classification function) of robust learning can be significantly larger than that of standard learning.

A comprehensive survey on the threat of adversarial attacks on deep learning models is given in [45].

5 Open Problems

Given the above overview of generalization error in deep learning, we provide here a list of open problems we have identified that we believe will have an important future impact on the field.

5.1 Problem 1: Generalization and Memorization

As reviewed in Sect. 4.1, understanding the capabilities and method of operation of deep neural networks requires a deeper understanding of the interplay between memorization and generalization. It has been shown that DNNs are powerful enough to memorize a random training dataset, yet with no actual generalization. It would be expected from a model that overfits any training data so well to obtain poor generalization, yet in practice DNNs generalize very well. It follows that currently existing theories are lacking since they are unable to explain this phenomenon, and a new comprehensive theory is required. Furthermore, an algorithm's ability to obtain a low generalization error strongly depends on the provided training dataset and not just on the model architecture. In order for effective learning to take place, the training dataset must be sufficiently large and well spread over the sample space in order to avoid the *curse of dimensionality*, a term widely used to refer to the need of a large amount of training data in high-dimensional problems. It follows that obtaining prior knowledge on the training dataset or the test dataset, e.g., the distributions from which they were drawn, would be highly beneficial in obtaining better generalization.

Several works make a relation between the generalization capabilities of DNNs and the underlying data model. For example, [46] examines which architecture is better for learning different functions. It is shown that deep neural networks, as opposed to shallow networks, are guaranteed to avoid the *curse of dimensionality* for an important class of problems: when the learned function is compositional. A thorough review of the abilities of shallow and deep neural networks to learn different kinds of compositional functions is done. Another recent work [47] examines the relationship between the classification performed by DNNs and the K-NN algorithm applied at the embedding space of these networks. The results suggest that a DNN generalizes by learning a new metric space adapted to the structure of the training dataset.

Following this track, we believe that better generalization bounds for learning algorithms are obtainable when an assumption on the data model is made. For instance, a sparsity assumption on the data model may be useful in this context, as can be observed in several prominent works such as [48–54].

5.2 Problem 2: Generalization and Robustness

Another prominent and interesting open problem is understanding the robustness properties of DNNs. A deep neural network is trained on a specific training dataset, which is sampled from some probability distribution. A good model, which has been adequately trained on a sufficiently large and balanced training dataset, is expected to generalize well on unseen test data which is drawn from the same distribution.

However, the question arises—how well would this DNN generalize to test data which was drawn from a different distribution? Can constraints on the relationship between the training distribution and the test distribution be imposed to guarantee good generalization? An example for this problem can be taken from the field of computer vision. Let us assume a DNN is trained to classify images which were taken of a scene in daylight. How well would this DNN generalize for images of the same scene taken at night? This is an important problem with implications for numerous applications, for example, autonomous vehicles. Networks trained to recognize issues on roads in sunny Silicon Valley may not work well in rainy and foggy London.

Another related and interesting question is how good is the cross-task adaptation of deep neural networks. How well can a DNN, which was trained to perform a certain task, perform another task? How beneficial would incremental learning [55] (a continuous "online" training of the model with new data) and transfer learning [56] (a wide term which in the implementational case is typically used to refer to the training of several layers of a network, which have been previously trained for performing a different task or on a different training dataset, for the purpose of performing a new task or to work well with a different data source) be in this case? How adaptive is a neural network between different tasks, and are there key design guidelines for obtaining better generalization, robustness and transferability for a network? How is the necessary information for good generalization embedded in the learned features, and in what sense is the network essentially learning a new metric? We refer the reader to [57] for a comprehensive survey of the field of transfer learning.

All of these questions represent highly important areas of research with substantial significance to the design of better DNN architectures with better generalization, robustness, and transferability.

5.3 Problem 3: Generalization and Adversarial Examples

A special case of the previous problem is the one of adversarial examples, which we have presented in Sect. 4.12. The counterintuitive vulnerability of DNNs to adversarial examples opens the door to a new angle in the research of the generalization of deep neural networks. It is of high importance to have a comprehensive theory dedicated to this type of examples. Some theories for explaining this phenomenon have been suggested, such as in [40, 41, 58]. However, this still remains an active field of research.

5.4 Problem 4: Generalization Error of Generative Models

An important lens through which the generalization capabilities of DNNs is examined is that of generative models. Generative models are models which are used to learn the underlying distribution from which the data is drawn, and thus manufacture more data from the same probability distribution, which can be used for training.

For example, Generative Adversarial Networks (GANs) [59] are a model which consists of two networks, a generative network G that captures the data distribution, and a discriminative network D that estimates the probability of a training sample being either genuine or fake (i.e., manufactured by G). In this minimax problem, the training phase results in G learning the underlying distribution of the training data. The work in [60] provided initial results regarding the generalization properties of GANs.

Another generative model which has gained much traction in the past years is the Variational Autoencoder (VAE) [61]. Classical VAEs are based on two neural networks, an encoder network and a decoder network. The encoder is used to learn a *latent variable*, from which the decoder generates a sample which is similar to the original input to the encoder, i.e., which is approximately drawn from the same distribution.

The question arises—which probability distributions can be learned under which conditions? How beneficial to the generalization capabilities would training on additional manufactured data be? These questions represent additional substantial paths for research with a promising impact on the field of deep learning.

5.5 Problem 5: Generalization Error and the Information Bottleneck

Recently, the information bottleneck has been introduced to explain generalization and convergence in deep neural networks [62, 63]. By characterizing the DNN *Information Plane*—the plane of the mutual information values that each layer preserves on the input and output variables—it is suggested that a network attempts to optimize the information bottleneck trade-off between compression and prediction for each layer.

It is also known that the information bottleneck problem is related to the information-theoretic noisy lossy source coding problem (a variation of the lossy source coding problem) [64]. In particular, a noisy lossy source coding problem with a specific loss function gives rise to the information bottleneck. Therefore, it is of interest to explore links between information theory, representation learning and the information bottleneck, in order to cast insights onto the performance of deep neural networks under an information-theoretic lens. Preliminary steps in this direction are taken in [65, 66].

6 Conclusions

Even though deep neural networks were shown to be a promising and powerful machine leaning tool which is highly useful in many tasks, the source of their capabilities remains somewhat elusive. Deep learning models are highly expressive, overparameterized, complex, non-convex models, which are usually trained (optimized) with a stochastic gradient method.

In this chapter, we reviewed the generalization capabilities of these models, shedding light on the reasons for their ability to generalize well from the training phase to the test phase, thus maintaining a low generalization error. We reviewed some of the fundamental works on this subject and also provided some more recent findings and theoretical explanations to the generalization characteristics of deep neural networks and the influence of different parameters on their performance.

We also reviewed various emerging open problems in deep learning, ranging from the interplay between robustness, generalization, and memorization, to robustness to adversarial attacks, the generalization error of generative models and the relation between the generalization error and the information bottleneck. These open problems require a deeper understanding to fully unlock the potential applicability of deep learning models in real environments. Beyond these open problems there are various other interesting learning settings that require much additional theoretical research, such as multi-modal learning, multi-task learning, incremental learning, the capacity of neural networks, the optimization of deep networks and more. While the current state of research in this field is promising, we believe that much room remains for further work to provide more comprehensive theories and a better understanding of this important subject as outlined throughout this chapter.

References

1. I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning* (MIT Press, 2016)
2. B.D. Haeffele, R. Vidal, Global optimality in neural network training, in *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, Honolulu, HI, USA, 21–26 July 2017, pp. 4390–4398
3. R. Vidal, J. Bruna, R. Giryes, S. Soatto, Mathematics of deep learning, in *Proceedings of the Conference on Decision and Control (CDC)* (2017)
4. V.N. Vapnik, A. Chervonenkis, The necessary and sufficient conditions for consistency in the empirical risk minimization method. Pattern Recogn. Image Anal. **1**(3), 260–284 (1991)
5. P.L. Bartlett, S. Mendelson, Rademacher and Gaussian complexities: risk bounds and structural results. J. Mach. Learn. Res. **3**(3), 463–482 (2002)
6. C. Zhang, S. Bengio, M. Hardt, B. Recht, O. Vinyals, Understanding deep learning requires rethinking generalization, in *ICLR* (2017)
7. D.A. McAllester, PAC-Bayesian model averaging, in *Proceedings of the twelfth annual Conference on Computational Learning Theory* (ACM, 1999), pp. 164–170
8. D.A. McAllester, Some PAC-Bayesian theorems. Mach. Learn. **37**(3), 355–363 (1999)
9. D. McAllester, Simplified PAC-Bayesian margin bounds, in *Learning Theory and Kernel Machines*, ed. by B. Schlkopf, M.K. Warmuth (Springer, Berlin, 2003), pp. 203–215
10. O. Bousquet, A. Elisseef, Stability and generalization. J. Mach. Learn. Res. **2**, 499–526 (2002)
11. H. Xu, S. Mannor, Robustness and generalization. Mach. Learn. **86**(3), 391–423 (2012)
12. K.P. Murphy, *Machine Learning: A Probabilistic Perspective*, 1st edn. (MIT Press, 2013)
13. S. Shalev-Shwartz, S. Ben-David, *Understanding Machine Learning: From Theory to Algorithms* (Cambridge University Press, New York, 2014)
14. N.S. Keskar, D. Mudigere, J. Nocedal, M. Smelyanskiy, P.T.P. Tang, On large-batch training for deep learning: generalization gap and sharp minima, in *ICLR* (2017)
15. S. Arora, R. Ge, B. Neyshabur, Y. Zhang, Stronger generalization bounds for deep nets via a compression approach, in *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*, Stockholmsmässan, Stockholm, Sweden, 10–15 July 2018, pp. 254–263
16. S. Ioffe, C. Szegedy, Batch normalization: accelerating deep network training by reducing internal covariate shift, in *Proceedings of the 32nd International Conference on International Conference on Machine Learning, ICML'15*, vol. 37 (2015). www.jmlr.org, pp. 448–456
17. N. Golowich, A. Rakhlin, O. Shamir, Size-independent sample complexity of neural networks, in Bubeck, S., Perchet, V., Rigollet, P., (eds.) *Proceedings of the 31st Conference on Learning Theory of Proceedings of Machine Learning Research*, vol. 75, PMLR, 06–09 July 2018, pp. 297–299

18. B. Neyshabur, S. Bhojanapalli, D. McAllester, N. Srebro, Exploring generalization in deep learning, in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*, 4–9 December 2017, Long Beach, CA, USA (2017), pp. 5949–5958
19. N. Harvey, C. Liaw, A. Mehrabian, Nearly-tight VC-dimension bounds for piecewise linear neural networks, in *Proceedings of the 2017 Conference on Learning Theory*, vol. 65 of Proceedings of Machine Learning Research, ed. by S. Kale, O. Shamir, Amsterdam, Netherlands, PMLR, 07–10 July 2017, pp. 1064–1068
20. P.L. Bartlett, V. Maiorov, R. Meir, Almost linear VC-dimension bounds for piecewise polynomial networks. Neural Comput. **10**(8), 2159–2173 (1998)
21. M. Anthony, P.L. Bartlett, *Neural Network Learning: Theoretical Foundations* (Cambridge University Press, New York, 2009)
22. B. Neyshabur, R. Tomioka, N. Srebro, Norm-based capacity control in neural networks, in *Proceedings of The 28th Conference on Learning Theory of Proceedings of Machine Learning Research*, vol. 40, ed. by P. Grnwald, E. Hazan, S. Kale, Paris, France, PMLR, 03–06 July 2015, pp. 1376–1401
23. B. Neyshabur, Z. Li, S. Bhojanapalli, Y. LeCun, N. Srebro, Towards understanding the role of over-parametrization in generalization of neural networks (2018). arXiv:1805.12076
24. A.R. Barron, J.M. Klusowski, Approximation and estimation for high-dimensional deep learning networks (2018). arXiv:1809.03090
25. B. Neyshabur, S. Bhojanapalli, N. Srebro, A PAC-Bayesian approach to spectrally-normalized margin bounds for neural networks, in *ICLR* (2018)
26. P.L. Bartlett, D.J. Foster, M.J. Telgarsky, Spectrally-normalized margin bounds for neural networks, in *Advances in Neural Information Processing Systems 30*, ed. by I. Guyon, U.V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Curran Associates, Inc., 2017), pp. 6240–6249
27. Dziugaite, G.K., Roy, D.M.: Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data, in *Proceedings of the Thirty-Third Conference on Uncertainty in Artificial Intelligence*, UAI 2016, pp. 11–15 (NSW, Sydney, 2017)
28. L. Dinh, R. Pascanu, S. Bengio, Y. Bengio, Sharp minima can generalize for deep nets, in *Proceedings of the 34th International Conference on Machine Learning of Proceedings of Machine Learning Research*, vol. 70, International Convention Centre, Sydney, Australia, PMLR, 06–11 August 2017, pp. 1019–1028
29. M. Hardt, B. Recht, Y. Singer, Train faster, generalize better: stability of stochastic gradient descent, in *ICML* (2016)
30. A.W. van der Vaart, J.A. Wellner, *Weak convergence and empirical processes: with applications to statistics*. Springer Series in Statistics. (Springer, 1996)
31. J. Sokolic, R. Giryes, G. Sapiro, M.R.D. Rodrigues, Robust large margin deep neural networks. IEEE Trans. Signal Process. **65**(16), 4265–4280 (2017)
32. W. Zhou, V. Veitch, M. Austern, R.P. Adams, P. Orbanz, Compressibility and generalization in large-scale deep learning (2018). arXiv:1804.05862
33. D. Soudry, E. Hoffer, M.S. Nacson, N. Srebro, The implicit bias of gradient descent on separable data, in *ICLR* (2018)
34. A. Brutzkus, A. Globerson, E. Malach, S. Shalev-Shwartz, Sgd learns over-parameterized networks that provably generalize on linearly separable data, in *ICLR* (2018)
35. H. Zhang, J. Shao, R. Salakhutdinov, Deep neural networks with multi-branch architectures are less non-convex (2018). arXiv:1806.01845
36. T.A. Poggio, K. Kawaguchi, Q. Liao, B. Miranda, L. Rosasco, X. Boix, J. Hidary, H. Mhaskar, Theory of deep learning iii: explaining the non-overfitting puzzle (2017). CoRR arXiv:1801.00173
37. E. Hoffer, I. Hubara, D. Soudry, Train longer, generalize better: closing the generalization gap in large batch training of neural networks, in *Advances in Neural Information Processing Systems 30*, ed. by I. Guyon, U.V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett (Curran Associates, Inc., 2017), pp. 1731–1741

38. J. Sokolic, R. Giryes, G. Sapiro, M.R.D. Rodrigues, Generalization error of invariant classifiers, in *Artificial Intelligence and Statistics* (2017), pp. 1094–1103
39. J. Bruna, S. Mallat, Invariant scattering convolution networks. IEEE Trans. Pattern Anal. Mach. Intell. **35**(8), 1872–1886 (2013)
40. C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, R. Fergus, Intriguing properties of neural networks, in *International Conference on Learning Representations* (2014)
41. I.J. Goodfellow, J. Shlens, C. Szegedy, Explaining and harnessing adversarial examples, in *ICLR* (2015)
42. R. Novak, Y. Bahri, D.A. Abolafia, J. Pennington, J. Sohl-Dickstein, Sensitivity and generalization in neural networks: an empirical study, in *ICLR* (2018)
43. D. Jakubovitz, R. Giryes, Improving DNN robustness to adversarial attacks using Jacobian regularization, in *The European Conference on Computer Vision (ECCV)* (2018)
44. L. Schmidt, S. Santurkar, D. Tsipras, K. Talwar, A. Madry, Adversarially robust generalization requires more data, in *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018*, 3–8 December 2018, Montréal, Canada. (2018) 5019–5031
45. N. Akhtar, A.S. Mian, Threat of adversarial attacks on deep learning in computer vision: a survey. IEEE Access **6**, 14410–14430 (2018)
46. T. Poggio, H. Mhaskar, L. Rosasco, B. Miranda, Q. Liao, Why and when can deep-but not shallow-networks avoid the curse of dimensionality: a review. Int. J. Autom. Comput. 1–17 (2017)
47. G. Cohen, G. Sapiro, R. Giryes, Dnn or k-nn: That is the generalize vs. memorize question (2018). arXiv:1805.06822
48. D. Vainsencher, S. Mannor, A.M. Bruckstein, The sample complexity of dictionary learning. J. Mach. Learn. Res. **12**, 3259–3281 (2011)
49. A. Jung, Y.C. Eldar, N. Grtz, Performance limits of dictionary learning for sparse coding. In: European Signal Processing Conference (EUSIPCO). (Sept 2014) 765–769
50. R. Gribonval, R. Jenatton, F. Bach, Sparse and spurious: dictionary learning with noise and outliers. IEEE Trans. Inf. Theory **61**(11), 6298–6319 (2015)
51. R. Gribonval, R. Jenatton, F. Bach, M. Kleinsteuber, M. Seibert, Sample complexity of dictionary learning and other matrix factorizations. IEEE Trans. Inf. Theory **61**(6), 3469–3486 (2015)
52. K. Schnass, Convergence radius and sample complexity of itkm algorithms for dictionary learning. Appl. Comput. Harmon. Anal. **45**(1), 22–58 (2018)
53. S. Singh, B. Pczos, J. Ma, On the reconstruction risk of convolutional sparse dictionary learning. In: AISTATS (2018)
54. V. Papyan, Y. Romano, M. Elad, Convolutional neural networks analyzed via convolutional sparse coding. J. Mach. Learn. Res. (JMLR) **18**(83), 1–52 (2017)
55. A. Gepperth, B. Hammer, Incremental learning algorithms and applications. In: European Symposium on Artificial Neural Networks (ESANN) (2016)
56. L. Torrey, J. Shavlik, Transfer Learning, in *Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques*, ed. by E. Soria, J. Martin, R. Magdalena, M. Martinez, A. Serrano. IGI Global (2009)
57. S.J. Pan, Q. Yang, A survey on transfer learning. IEEE Trans. Knowl. Data Eng. **22**(10), 1345–1359 (2010)
58. F. Tramer, N. Papernot, I. Goodfellow, D. Boneh, P. McDaniel, The space of transferable adversarial examples (2017). arXiv:1704.03453
59. I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in *Advances in Neural Information Processing Systems 27*, ed. by Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, K.Q. Weinberger (Curran Associates, Inc., 2014), pp. 2672–2680
60. S. Arora, R. Ge, Y. Liang, T. Ma, Y. Zhang, Generalization and equilibrium in generative adversarial nets (gans). In: ICML (2017)
61. D.P. Kingma, M. Welling, Auto-encoding variational bayes. In: ICLR (2014)

62. N. Tishby, N. Zaslavsky, Deep learning and the information bottleneck principle. In: 2015 IEEE Information Theory Workshop (ITW) (April 2015), pp. 1–5
63. R. Schwartz-Ziv, N. Tishby, Opening the black box of deep neural networks via information (2017). arXiv:1703.00810
64. T.M. Cover, J.A. Thomas, *Elements of Information Theory* (Wiley, New York, NY, USA, 2006)
65. M. Vera, L.R. Vega, P. Piantanida, Compression-based regularization with an application to multi-task learning. IEEE J. Sel. Top. Signal Process. **1–1** (2018)
66. P. Piantanida, L. Rey Vega, Information bottleneck and representation learning. In: Information-Theoretic Methods in Data Science (2018)

Deep Learning for Trivial Inverse Problems

Peter Maass

Abstract Deep learning is producing most remarkable results when applied to some of the toughest large-scale nonlinear problems such as classification tasks in computer vision or speech recognition. Recently, deep learning has also been applied to inverse problems, in particular, in medical imaging. Some of these applications are motivated by mathematical reasoning, but a solid and at least partially complete mathematical theory for understanding neural networks and deep learning is missing. In this paper, we do not address large-scale problems but aim at understanding neural networks for solving some small and rather naive inverse problems. Nevertheless, the results of this paper highlight the particular complications of inverse problems, e.g., we show that applying a natural network design for mimicking Tikhonov regularization fails when applied to even the most trivial inverse problems. The proofs of this paper utilize basic and well-known results from the theory of statistical inverse problems. We include the proofs in order to provide some material ready to be used in student projects or general mathematical courses on data analysis. We only assume that the reader is familiar with the standard definitions of feedforward networks, e.g., the backpropagation algorithm for training such networks. We also include—without proof—numerical experiments for analyzing the influence of the network design, which include comparisons with learned iterative soft-thresholding algorithm (LISTA).

1 Motivation and Outline

No matter in which field of science we are working and no matter which type of conferences or meetings we are attending, one of the hottest topics being discussed over the last few years is neural networks for large data applications. The success of

P. Maass (✉)
FB 3 Mathematik und Informatik, Zentrum für Technomathematik,
Universität Bremen, Bremen, Germany
e-mail: pmaass@math.uni-bremen.de

H. Boche et al. (eds.), *Compressed Sensing and Its Applications*,
Applied and Numerical Harmonic Analysis,
https://doi.org/10.1007/978-3-319-73074-5_6

such deep learning (DL) applications are stunning indeed in terms of their apparent success for large-scale real-life applications, see, e.g., [12, 18], but also with respect to the almost complete lack of theoretical justification.

While arguing and having experimental results is a sufficient foundation in some fields of sciences, it is not satisfactory in mathematics, where the concept of having a strict proof is essential. Some mathematical concepts for analyzing deep learning approaches are slowly emerging [1, 3, 7, 19, 23] and we want to add some basic results for the particular case of DL for inverse problems on the low and almost trivial side of complexity.

Our starting point is numerical experiments for linear systems $Ax = y$ with given noisy data y^δ. Surprisingly, even small two by two examples cannot be solved reliably by straightforward neural networks. Here, we employ minimal networks which are capable of learning a matrix–vector multiplication, i.e., such networks should be able to learn classical Tikhonov regularizers $(A^*A + \alpha I)^{-1}A^*$ or even better approximations. This small-scale setting allows a somewhat complete analysis of the neural network, in particular, we can prove the shortcomings of such neural networks if the condition number of the matrix and the noise level in the data are in a critical relation. The extension to linear systems of arbitrary dimension is straightforward.

On a conceptual level, when comparing inverse problems defined by analytical models $A : Z \to V$ with data-driven approaches such as deep learning, one would expect the data-driven approaches to have certain advantages if, e.g., A is an incomplete model of the underlying physical or engineering system or if the search for parameter x is actually restricted to a characteristic subset $Z_d \subset Z$, which, however, escapes precise mathematical modeling. In both situations, the missing information is implicitly contained in sufficiently large sets of experimentally measured pairs of data $(x^i, y^i)_{i=1,..,n}$. The assumption that A is only an incomplete model is not relevant for our small examples. However, we can test the potential of neural networks for learning the underlying structure or prior distribution of restricted parameter sets. Hence, we extend our experiments to the nonlinear problem of solving linear inverse problems with sparsity constraints [2, 4, 9, 15]. Here, we compare the results obtained by classical ISTA (iterated soft thresholding) with its learned counterpart LISTA [13]. The experimental results demonstrate that LISTA is not better than ISTA if one takes any set of sparse vectors as inputs. However, it performs significantly better if we assume structured sparsity of the inputs, i.e., if we restrict them to a low-dimensional subspace. In this case, the performance of ISTA does not change, but LISTA seems to unveil the underlying subspace and produces significantly improved results.

2 Basic Example

We consider the basic inverse problem given by an operator $A : Z \to V$ and some noisy data $y^\delta = Ax + \eta$. We further assume that a set of training data $(x^{(i)}, y^{(i)})_{i=1,..,n}$ is available for training a neural network. We will use this for training the forward operator, i.e., the set of $x^{(i)}$'s is input and $y^{(i)}$'s are the output, as well as for training the inverse problem, i.e., the set of $y^{(i)}$'s is input and $x^{(i)}$'s are the output.

To be precise, in our most basic example, we set $Z = V = I\!R^2$ and

$$A_\varepsilon = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 1+\varepsilon \end{pmatrix} .$$

This matrix has an orthogonal basis of eigenvectors $u_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \mathcal{O}(\epsilon^2)$, $u_2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix} + \mathcal{O}(\epsilon^2)$ and eigenvalues $\lambda_1 = 2 + \frac{\epsilon}{2} + \mathcal{O}(\epsilon^2)$, $\lambda_2 = \frac{\epsilon}{2} + \mathcal{O}(\epsilon^2)$. The ill-posedness of the problem—or rather the condition number of A—is controlled by $1/\varepsilon$; typical values are $\varepsilon = 10^{-k}$, $k = 0, ..., 10$.

As training data we draw n vectors $x^{(i)}$, $i = 1, .., n$, where each coefficient is i.i.d. $N(0, 1)$ normally distributed. The corresponding data vectors $Ax^{(i)}$ are corrupted with noise vectors $\eta^{(i)}$ where each coefficient of $\eta^{(i)}$ is drawn independently from a $N(0, \sigma^2)$ normal distribution, i.e.,

$$y^{(i)} = Ax^{(i)} + \eta^{(i)} . \tag{1}$$

For later use, we define $2 \times n$ matrices X (parameter matrix), Y (data matrix), and Θ (noise matrix) by storing column wise the vectors $x^{(i)}$, $y^{(i)}$, $\eta^{(i)}$, i.e.,

$$Y = AX + \Theta . \tag{2}$$

We now compare two methods for solving the inverse problem. The first one is the classical Tikhonov regularization, which only uses information about the operator A, i.e., for given data y we estimate the parameter x by $\hat{x}_{Tik} = (A^*A + \sigma^2 I)^{-1} A^* y$. The second inversion is based on a neural network Φ_W, which depends on a weight matrix W. W is obtained by training the neural network with respect to a so-called loss function. We use the standard least squares loss function $L_1(W) = \frac{1}{n} \sum_{i=1}^{n} \|\Phi_W(x^{(i)}) - y^{(i)}\|^2$ for training a net with parameters $W_{FP} = \text{argmin}\, L_1(W)$ for the forward problem and

$$L_2(W) = \frac{1}{n} \sum_{i=1}^{n} \|\Phi_W(y^{(i)}) - x^{(i)}\|^2 \tag{3}$$

for training $W_{IP} = \text{argmin}\, L_2(W)$ for solving the inverse problem. That is, this approach does not use any knowledge about the operator A. After training the inverse problem is solved by simply applying the inverse net $\hat{x}_\Phi = \Phi_{W_{IP}}(y)$.

The training is followed by an evaluation using a different set of test data. We compare these methods by computing the mean error for the test data

$$E_{Tik} := \frac{1}{n} \sum_{i=1}^{n} \|\hat{x}_{Tik}^{(i)} - x^{(i)}\|^2 \text{ resp. } E_\Phi := \frac{1}{n} \sum_{i=1}^{n} \|\hat{x}_\Phi^{(i)} - x^{(i)}\|^2$$

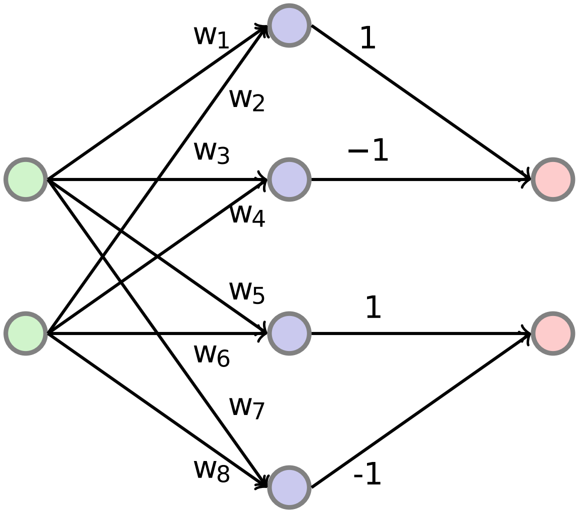


Fig. 1 The network design with eight parameters, setting $w_1 = -w_3 = w_{11}, w_2 = -w_4 = w_{12}, w_5 = -w_7 = w_{21}, w_6 = -w_8 = w_{22}$ yields a matrix–vector multiplication of the input

The design of the network is crucial. For our first tests, we use a minimal network which allows to reproduce a matrix–vector multiplication. Hence, the network is capable—in principle—to recover the Tikhonov regularization operator or even an improvement of it. Here, we use a network with a single hidden layer with 4 nodes and the standard $ReLU$-activation function, i.e., $ReLU(z) = \max\{z, 0\}$ (Rectified linear units, $z \in I\!R$). For the motivation of our network design, we observe $ReLU(z) - ReLU(-z) = z$. We restrict the eight weights connecting the input variables with the first layer by setting $w_1 = -w_3 = w_{11}, w_2 = -w_4 = w_{12}, w_5 = -w_7 = w_{21}, w_6 = -w_8 = w_{22}$ as depicted in Fig. 1. We obtain a neural network depending on four variables $w_{11}, w_{12}, w_{21}, w_{22}$ and the networks acts as multiplication with matrix $W = \begin{pmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{pmatrix}$ on the input vector $z = (z_1, z_2)$. We denote the output of such a neural network by $\phi_W(z) = Wz$.

The training of such a network for modeling the forward problem is equivalent (using the Frobenius norm for matrices) to minimizing the expected mean square error

$$\min_{W \in I\!R^{2\times 2}} \frac{1}{n} \sum_{i=1}^{n} \| Wx^{(i)} - y^{(i)} \|^2 = min_W \frac{1}{n} \| WX - Y \|^2 \tag{4}$$

and training a model for the inverse problem is done by

$$\min_{W} \frac{1}{n} \sum_{i=1}^{n} \| Wy^{(i)} - x^{(i)} \|^2 = \min_{W} \frac{1}{n} \| WY - X \|^2 \ . \tag{5}$$

In the next subsection, we report some numerical examples before we analyze these networks.

2.1 Testing Error Convergence for Various Values of ε

We train these networks using a set of training data $(x^{(i)}, y^{(i)})_{i=1,..n}$ with $n = 10.000$, i.e., $y^{(i)} = A_\epsilon x^{(i)} + \eta^{(i)}$. The network design with restricted coefficients as described above has four degrees of freedom $w = (w_{11}, w_{12}, w_{21}, w_{22})$. The corresponding loss function is minimized by a gradient descent algorithm, i.e., the gradient of the loss function with respect to w is computed by backpropagation [5, 20, 22]. We used 3.000 iterations (epochs) of this gradient descent for minimizing the loss function of a network for the forward operator using (4), respectively, for training a network for solving the inverse problem using (5). The MSE errors on the training data were close to zero in both cases.

After training, we tested the resulting networks by drawing $n = 10.000$ new data vectors $x^{(i)}$ as well as errors vectors $\eta^{(i)}$. The $y^{(i)}$ were computed as above.

In the following table, we show the resulting values using this set of test data $NMSE_{forward} = \frac{1}{n}\sum_{i=1}^{n} \|Wx^{(i)} - y^{(i)}\|^2$ for the network trained for the forward problem, respectively, $NMSE_{inverse}\frac{1}{n}\sum_{i=1}^{n} \|Wy^{(i)} - x^{(i)}\|^2$ for the network trained for the inverse problem.

Error/choice of ε	1	0.1	0.01	0.0001
NMSE (direct problem)	0.002	0.013	0.003	0.003
NMSE (inverse problem)	0.012	0.8	10	10

The errors of the inverse net are large and the computed reconstructions with the test data are meaningless. We have also evaluated the mean squared errors after each iteration of the training as depicted in Fig. 2a, b. Here, the values of the errors are shown for the first 3.000 iterates and for data produced with different values of ϵ.

We observe that the training of the forward operator produces reliable results as well does the network for the inverse problem with $\varepsilon \geq 0.1$. However, training a network for the inverse problem with an ill-conditioned matrix A_ϵ with $\epsilon \leq 0.01$ fails.

This is confirmed by analyzing the values of w and of the resulting matrix W after training: We would assume that in training the forward problem we produce values for w such that $W \sim A_\epsilon$ and that training the inverse problems leads to $W \sim (A^*A + \sigma^2)^{-1}A^*$. For the forward problem, the difference between W and A is of the order 10^{-3} or below, but for $\epsilon \leq 0.01$ the training of the inverse problem leads to a matrix, which has no similarity with the Tikhonov regularized inverse. Using a

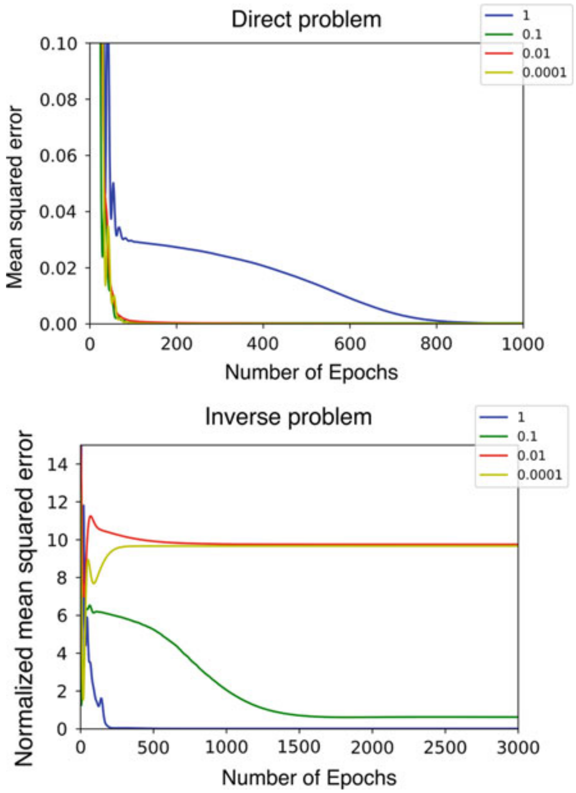


Fig. 2 History of the values of the loss function during training of the forward map (left, **a**) and the inverse problem (right, **b**). ϵ is set to $1, 0.1, 0.01, 0.0001$

network with a single internal layer, but with more nodes and no restriction on the structure of the weights, did not yield any significant improvements.

3 Analysis of Trivial Neural Networks for Inverse Problems

The numerical examples indicate that training even a most simple neural network for a well-posed matrix–vector multiplication (forward operator) yields good results. However, the natural approach for training a network for an inverse problem by reversing inputs and outputs fails in certain cases. In this section, we aim to analyze why this approach has its limitations for inverse problems.

3.1 Matrix Case

We consider the training of the trivial network, see Fig. 1, for solving an inverse problem with matrix $A = A_\epsilon$. That is, $y^{(i)} = Ax^{(i)} + \eta^{(i)}$ are inputs to the network and the loss function is the mean squared error between the outputs of the network and the exact solutions $x^{(i)}$.

Hence, in this section, we analyze the special case, where the application of the neural network is strictly equivalent to a matrix–vector multiplication. In our test example, this refers to restricting the coefficients of the neural network to four coefficients $\tilde{w}_{11}, \tilde{w}_{12}, \tilde{w}_{21}, \tilde{w}_{22}$ and setting $w_1 = -w_3 = \tilde{w}_{11}$, $w_2 = -w_4 = \tilde{w}_{12}$, $w_5 = -w_7 = \tilde{w}_{21}$, $w_6 = -w_8 = \tilde{w}_{22}$. The output of the network yields

$$\phi(W, y) = Wy \quad \text{where} \quad W = \begin{pmatrix} \tilde{w}_{11} & \tilde{w}_{12} \\ \tilde{w}_{21} & \tilde{w}_{22} \end{pmatrix} .$$

Training of the network is equivalent to determining a matrix W which minimizes

$$\min_W \frac{1}{n} \| WY - X \|^2 .$$

Analyzing this discrepancy functional is the classical situation in statistical inverse problems theory [8, 16, 21], where training W is equivalent to determining the MAP (maximum a posteriori) estimator.

The optimal W is obtained as

$$W^T = (YY^T)^{-1} YX^T . \tag{6}$$

We analyze W by using (1) and obtain the following expression for YY^T:

$$YY^T = (AX + \Theta)(AX + \Theta)^T = AXX^T A^T + AX\Theta^T + \Theta X^T A^T + \Theta\Theta^T . \tag{7}$$

Lemma 1 *For a given A, we denote by Y the matrix of training data, by X the matrix containing the parameters in the training set, and by Θ a noise matrix as defined in Sect. 2. Then*

$$\frac{1}{n} YY^T = (AA^T + \sigma^2 I) + R \quad \textit{with} \quad R = AB_1 A^T + B_2 + AB_3 + B_4 A^T \quad ,$$

where $B_1 = \frac{1}{n} XX^T - I$, $B_2 = \frac{1}{n}\Theta\Theta^T - \sigma^2 I$, $B_3 = \frac{1}{n} X\Theta^T$, $B_4 = \frac{1}{n}\Theta X^T$.

Then

$$0 = I\!E(B_1) = I\!E(B_2) = I\!E(B_3) = I\!E(B_4)$$

$$var(B_1) = \frac{1}{n}\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \ , \ var(B_2) = \frac{\sigma^4}{n}\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \ , \ var(B_3) = var(B_4) = \frac{\sigma^2}{n}\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \ .$$

Remark 1 We will use some basic results for normally distributed random variables $Z \sim N(0, \sigma^2)$, see, e.g., [10, 11]:

$$I\!E(Z) = 0 \ , \ var(Z) = I\!E(Z^2) = \sigma^2 \ , \ var(Z^2) = I\!E(Z^4) - I\!E(Z^2)^2 = 2\sigma^4 \ .$$

If $Z^{(i)}$ are i.i.d. (not necessarily normally distributed), then

$$I\!E(\sum_{i=1}^{n} Z^{(i)}) = \sum_{i=1}^{n} I\!E(Z^{(i)}) \ , \ var(\sum_{i=1}^{n} Z^{(i)}) = \sum_{i=1}^{n} var(Z^{(i)}) \ .$$

If $Z, \tilde{Z}$ are i.i.d. $N(0, \sigma^2)$ random variables, then

$$I\!E(Z\tilde{Z}) = 0 \ , \ var(Z\tilde{Z}) = I\!E(Z^2\tilde{Z}^2) = I\!E(Z^2)I\!E(\tilde{Z}^2) = \sigma^4 \ .$$

Proof By definition we have $\eta_1^{(i)}, \eta_2^{(i)}, i = 1, .., n$ are i.i.d. $N(0, \sigma^2)$ random variables and Θ is the corresponding $2 \times n$ matrix. Then

$$\Theta\Theta^T = \begin{pmatrix} \sum_{i=1}^n (\eta_1^{(i)})^2 & \sum_{i=1}^n \eta_1^{(i)}\eta_2^{(i)} \\ \sum_{i=1}^n \eta_1^{(i)}\eta_2^{(i)} & \sum_{i=1}^n (\eta_2^{(i)})^2 \end{pmatrix} = n(\sigma^2 I + B_2)$$

with $B_2 = \frac{1}{n}\begin{pmatrix} -n\sigma^2 + \sum_{i=1}^n (\eta_1^{(i)})^2 & \sum_{i=1}^n \eta_1^{(i)}\eta_2^{(i)} \\ \sum_{i=1}^n \eta_1^{(i)}\eta_2^{(i)} & -n\sigma^2 + \sum_{i=1}^n (\eta_2^{(i)})^2 \end{pmatrix}$.

For fixed n, we obtain

$$I\!E(-n\sigma^2 + \sum_{i=1}^{n} (\eta_1^{(i)})^2) = -n\sigma^2 + \sum_{i=1}^{n} I\!E\left((\eta_1^{(i)})^2\right) = 0 \quad \text{and}$$

$$var\left(-\sigma^2 + \frac{1}{n}\sum_{i=1}^{n} (\eta_1^{(i)})^2\right) = \frac{1}{n^2}\sum_{i=1}^{n} var\left((\eta_1^{(i)})^2\right) = \frac{2}{n}\sigma^4 \ .$$

Similarly, by using the rules above and by defining the variance of a matrix componentwise, we obtain

$$I\!E(\frac{1}{n}\Theta\Theta^T) = \sigma^2 I \ , I\!E(B_2) = 0 \text{ and } \ var(B_2) = var\left(\frac{1}{n}\Theta\Theta^T\right) = \frac{\sigma^4}{n}\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \ .$$

Similarly, with $x_1^{(i)} \sim N(0, 1)$, we obtain $XX^T = n(I + B_1)$ and

$$I\!E(\frac{1}{n}XX^T) = I \ \text{ and } \ var(\frac{1}{n}XX^T) = \frac{1}{n}\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \ .$$

Setting $B_3 := \frac{1}{n} X\Theta^T$ and $B_4 := B_3^T$ yields

$$I\!E(X\Theta^T) = I\!E(\Theta X^T) = 0 \text{ and } var(B_3) = var(B_4) = \frac{\sigma^2}{n}\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} .$$

We now analyze $W^T = (\frac{1}{n} Y Y^T)^{-1} \frac{1}{n} Y X^T$. With the notation for $B_1, .., B_4$ as in the lemma above and assuming that the reminder term R is small enough, we obtain by using Neumann series

$$(\frac{1}{n} Y Y^T)^{-1} = (AA^T + \sigma^2 I)^{-1} + (AA^T + \sigma^2 I)^{-1} \sum_{k \geq 1} Q^k \ . \tag{8}$$

with

$$Q = (AA^T + \sigma^2 I)^{-1} R = (AA^T + \sigma^2)^{-1}(AB_1A^T + B_2 + AB_3 + B_4A^T) \ . \tag{9}$$

This gives the expected result, namely, that training our specific network for the inverse problems tries to mimic Tikhonov regularization. However, this is only valid if Q is indeed small. We analyze the norms of Q and R in a series of lemmata. First, we analyze the deterministic part of Q and R.

Lemma 2 *Let $A = A_\epsilon$ be defined as above. Then*

$$\|(AA^T + \sigma^2 I)^{-1}\| = \mathcal{O}\left(\frac{1}{\epsilon^2 + \sigma^2}\right) = \mathcal{O}\left(\min(1/\epsilon^2, 1/\sigma^2)\right) \ ,$$
$$\|(AA^T + \sigma^2 I)^{-1} A\| = \mathcal{O}\left(\frac{\epsilon}{\epsilon^2 + \sigma^2}\right) \ .$$

Proof We use the specific form of our matrix A and the values of its eigenvalues as computed above. Classical arguments using the singular value decomposition of A show that the eigenvalues ν_1, ν_2 of $(AA^T + \sigma^2 I)^{-1}$, resp. of $(AA^T + \sigma^2 I)^{-1}A$, are given by

$$\nu_1 = \frac{1}{\lambda_1^2 + \sigma^2} = \frac{1}{4} + \mathcal{O}(\epsilon + \sigma^2) \text{ and } \nu_2 = \frac{1}{\lambda_2^2 + \sigma^2} = \frac{1}{\epsilon^2/4 + \sigma^2}(1 + \mathcal{O}(\epsilon)) \, ,$$

$$\text{resp. } \nu_1 = \frac{\lambda_1}{\lambda_1^2 + \sigma^2} = \frac{1}{2} + \mathcal{O}(\epsilon + \sigma^2) \text{ and } \nu_2 = \frac{\lambda_2}{\lambda_2^2 + \sigma^2} = \frac{\epsilon/2}{\epsilon^2/4 + \sigma^2}(1 + \mathcal{O}(\epsilon)) \, .$$

For symmetric matrices, the spectral radius is equivalent to matrix norms. Hence, for small values of ϵ and σ, we obtain the asymptotic estimate of the norm of these matrices, which is determined by the value of the second eigenvalue λ_2 of A, i.e.,

$$\|(AA^T+\sigma^2 I)^{-1}\| = \mathcal{O}\left(\frac{1}{\epsilon^2+\sigma^2}\right) = \mathcal{O}\left(\min(1/\epsilon^2, 1/\sigma^2)\right) ,$$
$$\|(AA^T+\sigma^2 I)^{-1}A\| = \mathcal{O}\left(\frac{\epsilon}{\epsilon^2+\sigma^2}\right) .$$

Estimating the spectral radius of products of normally distributed random matrices with variable variances is the topic on ongoing research, see, e.g., [17, 24]. Motivated by the results and conjectures stated in these papers, we take the expectation values of the individual entries of our random matrices as representative values for their respective norms. That is, we take the componentwise estimates as an estimate for the spectral radius of B. These values are the square roots of the variances computed in Lemma 3.1. and we obtain the following corollary.

Corollary 1 *We define the "pseudo spectral" radius $\tilde{\rho}(B)$ for $B \in \{B_i, i = 1, .., 4\}$, where B_i is defined as above by*

$$\tilde{\rho}(B) = max_{i,j}\left(\sqrt{I\!E(b_{ij}^2)}\right) ,$$

where $b_{i,j}$ are the entries of the matrix B. Then Lemma 3.1 implies

$$\tilde{\rho}(B_1) = \sqrt{\frac{2}{n}} , \quad \tilde{\rho}(B_2) = \sqrt{\frac{2\sigma^4}{n}} , \quad \tilde{\rho}(B_3) = \sqrt{\frac{2\sigma^2}{n}} , \quad \tilde{\rho}(B_4) = \sqrt{\frac{2\sigma^2}{n}} .$$

Combining the last two statements allows us to obtain an estimate of the norm of the four terms of Q in (9), e.g., we obtain for two of these expressions:

$$\|(AA^T+\sigma^2 I)^{-1}AB_1A^T\| \leq \|(AA^T+\sigma^2 I)^{-1}A\|\|B_1\|\|A^T\| = \mathcal{O}\left(\frac{\epsilon}{\sqrt{n}(\epsilon^2+\sigma^2)}\right)$$

$$\|(AA^T+\sigma^2 I)^{-1}B_4A^T\| \leq \|(AA^T+\sigma^2 I)^{-1}\|\|B_4\|\|A^T\| = \mathcal{O}\left(\frac{\sigma}{\sqrt{n}(\epsilon^2+\sigma^2)}\right) .$$

Similarly, we obtain estimates for the other terms, hence, componentwise

$$I\!E(\|Q\|) = \mathcal{O}\left(\frac{\epsilon+\sigma}{\sqrt{n}(\epsilon^2+\sigma^2)}\right) = \mathcal{O}\left(\frac{1}{\sqrt{n}(\epsilon+\sigma)}\right) . \tag{10}$$

As we will see in the next lemma, this is actually a sharp estimate under rather weak assumptions. We only need to ensure that the eigenvectors of the random matrices $B_1, .., B_4$ are in random position and do not align with the eigenvectors of A. We state the result only for the first term in the expression for Q the estimates for the other terms follows equivalently.

Lemma 3 *Let $\lambda_1, \lambda_2, u_1, u_2$, resp. ν_1, ν_2, v_1, v_2, denote eigenvalues and normalized eigenvectors of A, resp. B_1, such that the spectral radius is given by $\rho(A) = \lambda_1$, resp.*

$\rho(B_1) = \nu_1$. *Assume that there exist constants* $c, \tilde{c} > 0$ *such that*

$$| < u_1, v_1 > | \geq c \ , \ | < u_1, v_2 > | \geq c \ \ \textit{and} \ \ |1 - \nu_2/\nu_1| \geq \tilde{c} \ .$$

Then

$$\begin{aligned}\|(AA^T + \sigma^2 I)^{-1} A B_1 A^T\| &\geq \tilde{c} c^2 \rho(AA^T + \sigma^2 I)^{-1} A) \rho(B_1) \rho(A) \\ &= \mathcal{O}\left(\frac{\epsilon}{\sqrt{n}(\epsilon^2 + \sigma^2)}\right) \ .\end{aligned}$$

Proof A and B_1 are symmetric matrices; hence, the respective eigenvalues are real, the eigenvectors form an orthonormal basis, and u_1, u_2 are also eigenvectors of $(AA^T + \sigma^2 I)^{-1} A$. However, the spectral radius of this matrix is given by the eigenvalue for u_2.

We consider $C = (AA^T + \sigma^2 I)^{-1} A B_1 A^T$ and obtain a lower estimate by computing Cu_1, where u_1 is an eigenvector corresponding to the largest eigenvalue of A, i.e., $Au_1 = \lambda_1 u_1 = \rho(A) u_1$. We use the expansion $u_1 =< u_1, v_1 > v_1 + < u_1, v_2 > v_2$ and obtain

$$B_1 u_1 = \nu_1 < u_1, v_1 > v_1 + \nu_2 < u_1, v_2 > v_2 \ .$$

We now expand $v_1, \ v_2$ in $\{u_1, \ u_2\}$ and observe the orthogonality of the eigenfunctions implies $< u_1, v_2 >< v_2, u_2 > = - < u_1, v_1 >< v_1, u_2 >$. Rearranging some terms, we finally obtain

$$\|Cu_1\| \geq | < Cu_1, u_2 > | \geq \tilde{c} c^2 \rho(AA^T + \sigma^2 I)^{-1} A) \rho(B_1) \rho(A) \ .$$

This leads to the final result on the structure of W, which just summarizes the previous statements.

Theorem 1 *Let* W, Q *be defined as in Lemma 3.1 and (9). If* $\|Q\| < 1$, *then*

$$\begin{aligned}W^T &= (\frac{1}{n} Y Y^T)^{-1} \frac{1}{n} Y X^T \\ &= (AA^T + \sigma^2 I)^{-1} A + (AA^T + \sigma^2 I)^{-1} \sum_{k \geq 1} Q^k A (I + B_1) + (AA^T + \sigma^2 I)^{-1} A\end{aligned}$$

and $\mathbb{E}(W) = (A^T A + \sigma^2 I)^{-1} A^T$.

The coefficients of the matrix Q *satisfy componentwise*

$$\mathbb{E}(\|Q\|) = \mathcal{O}\left(\frac{1}{\sqrt{n}(\epsilon + \sigma)}\right) \ .$$

Proof By definition $YX^T = A + AB_1$, hence, the claim for W^T follows directly from (9). The second part of the theorem follows from (10). □

Table 1 Analysis of absolute error rates. Errors are computed as $err = 1/n \sum_{i=1}^{n} \|\Phi_W(y_i) - x_i\|^2$

Error for $\sigma \backslash \varepsilon$	1	$1e-1$	$1e-2$	$1e-3$	$1e-4$
1	0.00889	0.01063	0.01082	0.01090	0.01076
1e-1	0.00548	0.06605	0.08061	0.08008	0.07962
1e-2	0.00058	0.03269	0.64470	0.79562	0.80357
1e-3	0.00006	0.00334	0.30427	6.36951	7.96213
1e-4	0.00001	0.00033	0.03193	3.03402	64.84077
1e-5	0.00000	0.00003	0.00320	0.31916	30.81500

Table 2 Analysis of relative error rates. Errors are computed as $err = 1/n \sum_{i=1}^{n} \|\Phi_W(y_i) - x_i\|^2/\|x_i\|^2$

Error for $\sigma \backslash \varepsilon$	1	$1e-1$	$1e-2$	$1e-3$	$1e-4$
1	0.01881	0.02704	0.02682	0.02708	0.02682
1e-1	0.00220	0.01625	0.11276	0.15959	0.16173
1e-2	0.00022	0.00166	0.01590	0.15882	1.12600
1e-3	0.00002	0.00016	0.00159	0.01595	0.16058
1e-4	0.00000	0.00002	0.00016	0.00159	0.01601
1e-5	0.00000	0.00000	0.00002	0.00016	0.00158

Remark 2 The neural network will train a matrix $W = (YY^T)^{-1}YX^T$, whose expectation values coincide with the Tikhonov regularizers $= (AA^T + \sigma^2)^{-1}A$. This is not a surprising result since T coincides with the classical MAP estimator of statistical inverse problems, see [16]. However, analyzing its variance $\mathbb{E}(\|W - T\|^2)$ we are lead to analyze the spectral radius or norm of Q, which determines the convergence of the Neumann series in (8). Its behavior is characterized in (10), which reflects the ill-posedness of the problem. No matter how many data points we have (fixed n), the deviation of W from T will be arbitrarily large if ϵ and σ are both small. Of course, we can also give this a positive meaning, e.g., the noise level acts as a regularizer, large σ yields more stable matrices W.

This 2×2 problem is only a toy problem. If dealing with approximations to infinite-dimensional inverse problems, then ϵ will tend to zero with increasing accuracy of any numerical approximation. In this case, we cannot expect that simple neural networks as described above will yield meaning full results.

The above-described derivations are validated by numerical experiments with variable ϵ and σ (Tables 1 and 2).

These tests demonstrate that there exists a critical linear relation between ϵ and σ, which leads to large error rates. For fixed ϵ, the error rates can get smaller if the noise level σ is increased. This might be counterintuitive, since large noise levels should lead to less quality in the reconstructions. This is actually also the case here, but the table states the deviation from the Tikhonov matrix T, which depends itself on σ.

4 Further Numerical Tests

The findings of the previous section should be understood as a warning that inverse problems do have their rather specific complications and just applying seemingly suitable networks for solving inverse problems can fail miserably. Hence, investigating neural networks specifically for inverse problems makes sense and, of course, this has been done for all kinds of inverse problems already. Some most remarkable papers address unrolling of iteration schemes for solving inverse problems [1, 13], optimizing proximal mappings [14], or on constructing suitable penalty terms by neural networks [6, 7]. However, they lack a thorough convergence analysis.

Nevertheless, one has to admit that these more advanced schemes perform well for large-scale applications and also for our small toy problem. We just report on some of our numerical experiments for sparsity constrained matrix equations using ISTA (iterated soft thresholding) and LISTA (learned ISTA). We start with defining the algorithms.

ISTA: Let $y \in I\!R^{d_2}$, $A \in I\!R^{d_2 \times d_1}$ be given, choose λ, α and set $x^0 = 0$.

For k = 1, ... do

$$x^k = \mathcal{S}_\alpha\left((I - \lambda A^T A)x^{k-1}\right)$$

until *stopping criterion*.

Here, $\mathcal{S}_\alpha(x)$ is defined componentwise by

$$\mathcal{S}_\alpha(x)_j = sign(x_j)\max\{|x_j| - \alpha, 0\}.$$

One can reformulate the iteration step as

$$x^k = \mathcal{S}_\alpha\left((I - \lambda A^T A)x^{k-1} + \lambda A^T Ay\right) = \mathcal{S}_\alpha\left(Wx^{k-1} + By\right), \tag{11}$$

where $B = \lambda A^T \in I\!R^{d_1 \times d_2}$ and $W = I - BA \in I\!R^{d_1 \times d_1}$.

This is exactly the structure of the computations performed by a neural network $\Phi_{W,B}$ with activation function $\mathcal{S}_\alpha$. Hence, one can determine two matrices W, B by training a fully connected feedforward neural network with K internal layers using the loss function L_2 for the inverse problem as above. Applying the trained network on an input y^δ, one expects $\Phi_{W,B}(y^\delta) \sim x$.

LISTA: Let $\Phi_{W,B}$ denote a fully connected feedforward network with K internal layers with d_1 nodes in each layer. The linear maps as in (11) are optimized during training , $\mathcal{S}_\alpha$ is kept fixed as activation function, and the identity is used as output layer. Let $(y^{(i)} \in I\!R^{d_2}, x^{(i)})_{i=1,..,n}$, $y^{(i)}, x^{(i)} \in I\!R^d_1$ denote a set of training data and determine

$$(W^*, B^*) = \text{argmin}_{W,B} \sum_{i=1,n} \|\Phi_{W,B}(y^{(i)} - x^{(i)}\|^2 .$$

Table 3 Comparison of error rates for ISTA and LISTA

	ISTA	LISTA
MSE	0.6672	1.6274

Table 4 Comparison of error rates for ISTA and LISTA with low-dimensional input data

Sparsity /MSE	ISTA	LISTA
70% (unstructured)	0.6578	0.6642
70% (structured)	0.5439	0.0786
80% (unstructured)	0.4471	0.4778
80% (structured)	0.3083	0.0196

After training, we use a separate set of test data for evaluation by the same procedure as above. ISTA is initialized with $\alpha = \sigma$ and $\lambda = 0.01$, and the stopping criterion is $x^k - x^{k-1} \leq 10^{-6}$. Even for the small 2×2 example, this leads to 100 iterations or more before convergence. For LISTA, we set $K = 10$, i.e., it optimizes 10 iterations. Using more internal layers does not improve performance. Using uniformly sampled input data, i.e., the data are drawn uniformly form a ball around 0, the performance of LISTA is stable but does not improve, when compared with ISTA (Table 3).

This comparison is somewhat unfair, as it compares a fully converged ISTA with many iterations with a partially converged LISTA mimicking K iterations. Other papers, see [13], compare LISTA with K internal layers with ISTA with K steps, which—at least for K not too large—shows an advantage for LISTA.

In order to exploit the potential of neural networks for discovering prior distributions of the training data, we have changed the setup of the experiment. We did choose 10-dimensional input and output vectors and a singular matrix A. Moreover, the test data were drawn from a 2-or 3−dimensional linear subspace. In this case, ISTA performs as before, which is not surprising. ISTA is built just using A and does not incorporate any knowledge about the input data. LISTA, however, performs much better, as it seems to discover the underlying low-dimensional structure.

Let $A = \begin{pmatrix} 1 & \cdots & 1 \\ \vdots & & \vdots \\ 1 & \cdots & 1 + \epsilon \end{pmatrix}$ and $\epsilon = 1$. Let $x \in [0, 1]^{10}$ be sparse. The feedforward neural network is trained for 50 epochs and has $K = 15$ layers. We observe the following error rates (Table 4).

Acknowledgements The author acknowledges the final support provided by the Deutsche Forschungsgemeinschaft (DFG) under grant GRK 2224/1 "Pi3: Parameter Identification—Analysis, Algorithms, Applications". The numerical examples were done by Hannes Albers and Alexander Denker, who in particular designed the experiment with sparse input data. The statistical analysis was supported by Max Westphal. Furthermore, the author wants to thank Carola Schönlieb for her hospitality; part of the paper was written during the authors sabbatical in Cambridge. Finally, the author wants to thank a reviewer for careful reading and suggesting several improvements.

References

1. J. Adler, O. Öktem, Solving ill-posed inverse problems using iterative deep neural networks. Inverse Probl. **33**(12), 124007 (2017)
2. J. Bioucas-Dias, M. Figueiredo, A new twist: two-step iterative shrinkage/thresholding algorithms for image restoration. **16**, 2992–3004 (2008)
3. H. Bölcskei, P. Grohs, G. Kutyniok, P. Petersen, Optimal approximation with sparsely connected deep neural networks. CoRR abs/1705.01714abs/1705.01714 (2017)
4. T. Bonesky, K. Bredies, D.A. Lorenz, P. Maass, A generalized conditional gradient method for nonlinear operator equations with sparsity constraints. Inverse Probl. **23**(5), 2041 (2007)
5. R.H. Byrd, G.M. Chin, J. Nocedal, W. Yuchen, Sample size selection in optimization methods for machine learning. Math. Programm. **134**(1), 127–155 (2012)
6. Y. Chen, T. Pock, Trainable nonlinear reaction diffusion: a flexible framework for fast and effective image restoration. IEEE Trans. Pattern Anal. Mach. Intell. **39**(6), 1256–1272 (2017)
7. C. Chung Van, J.C. De los Reyes, C.B. Schoenlieb, Learning optimal spatially-dependent regularization parameters in total variation image denoising. Inverse Probl. **33**(7), 074005 (2017)
8. D. Colton, H. Engl, A.K. Louis, J. McLaughlin, W. Rundell, *Surveys on Solution Methods for Inverse Problems* (Springer, 2000). http://www.deeplearningbook.org
9. I. Daubechies, M. Defrise, C. De Mol, An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. Commun. Pure Appl. Math. **57**(11), 1413–1457 (2004)
10. A. Edelman, B.D. Sutton, Y. Wang, *Random Matrix Theory, Numerical Computation and Applications*
11. A. Edelman, N.R. Rao, Random matrix theory. Acta Numer. **14**, 233–297 (2005)
12. I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning* (MIT Press, 2016). http://www.deeplearningbook.org
13. K. Gregor, Y. LeCun, Learning fast approximations of sparse coding, in *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10 (Omnipress, USA, 2010), pp. 399–406
14. A. Hauptmann, F. Lucka, M. Betcke, N. Huynh, J. Adler, B. Cox, P. Beard, S. Ourselin, S. Arridge, Model based learning for accelerated, limited-view 3D photoacoustic tomography. IEEE Trans. Med. Imaging (2018). In Press
15. B. Jin, P. Maass, Sparsity regularization for parameter identification problems. Inverse Probl. **28**(12), 123001 (2012)
16. J. Kaipio, E. Somersalo, *Statistical and Computational Inverse Problems* (Springer, 2005)
17. R. Latala, Some estimates of norms of random matrices. Proc. Am. Math. Soc. **133**(5), 1273–1282 (2005)
18. Y. LeCun, Y. Bengio, G. Hinton, Deep learning. Nature **521**(7553), 436–444 (2015)
19. S. Mallat, Understanding deep convolutional networks. CoRR, abs/1601.04920 (2016)
20. J. Martens, I. Sutskever, *Training Deep and Recurrent Networks with Hessian-Free Optimization* (Springer, Berlin, Heidelberg, 2012), pp. 479–535
21. J.L. Mueller, S. Siltanen, *Linear and Nonlinear Inverse Problems with Practical Applications* (SIAM, 2012)
22. D.E. Rumelhart, G.E. Hinton, R.J. Williams, *Neurocomputing: Foundations of Research. Chapter Learning Representations by Back-propagating Errors* (MIT Press, Cambridge, MA, USA, 1988), pp. 696–699
23. M. Unser (2018) A representer theorem for deep neural networks. ArXiv e-prints
24. R. van Handel (2015) On the spectral norm of Gaussian random matrices. ArXiv e-prints

Oracle Inequalities for Local and Global Empirical Risk Minimizers

Andreas Elsener and Sara van de Geer

Abstract The aim of this chapter is to provide an overview of general frameworks used to derive (sharp) oracle inequalities. Two extensions of a general theory for convex norm penalized empirical risk minimizers are summarized. The first one is for convex nondifferentiable loss functions. The second is for nonconvex differentiable loss functions. Theoretical understanding is required for the growing number of algorithms in statistics, machine learning, and, more recently, deep learning that are based on (combinations of) these types of loss functions. To motivate the importance of oracle inequalities, the problem of model misspecification in the linear model is first discussed. Then, the sharp oracle inequalities are stated. Finally, we show how to apply the general theory to problems from regression, classification, and dimension reduction.

1 Introduction

1.1 *Model Misspecification*

Often, one is faced with the problem of choosing a model for a given set of data and for a given purpose (typically prediction for new data points). This choice then influences the types of estimators and statistical analyses one carries out. The cases where a chosen model exactly explains the data are very rare. Nevertheless, despite the "wrong model", one fortunately observes that the chosen model is not completely

A. Elsener (✉) · S. van de Geer
ETH Zürich, Seminar für Statistik, Rämistrasse 101, 8092 Zürich, Switzerland
e-mail: elsener@stat.math.ethz.ch

S. van de Geer
e-mail: geer@stat.math.ethz.ch

H. Boche et al. (eds.), *Compressed Sensing and Its Applications*,
Applied and Numerical Harmonic Analysis,
https://doi.org/10.1007/978-3-319-73074-5_7

far-off the underlying truth. Our interest lies in the quantification of this intuition. The goal is to discuss new and already known theoretical tools to analyze specific classes of statistical estimation problems under possible model misspecification.

Before providing the exact framework, it is worth discussing the linear model as done in Rigollet's lecture notes [26]. The situation described in the previous paragraph should become more tangible. The measurements/data consist of n pairs (Y_i, X_i) where $Y_i \in \mathbb{R}$ is the response and $X_i \in \mathbb{R}^p$ a p-dimensional row vector. For some unknown but fixed real-valued function f, the dependency is then assumed to be

$$Y_i = f(X_i) + \varepsilon_i,$$

where for all $i = 1, \ldots, n$ the errors ε_i are assumed to be i.i.d. and independent of X_i. One way to model the function f is to assume that it is linear in the parameters $(\beta_1^0, \ldots, \beta_p^0) =: \beta^0 \in \mathbb{R}^p$, where the vector β^0 is assumed to be a column vector throughout the chapter:

$$Y_i = X_i\beta^0 + \varepsilon_i.$$

This is usually not exactly true but it may be a good approximation. After estimating β^0, it provides among other properties a powerful tool to predict the outcome for new data points. An attempt to understand and explain why it might be sensible to use a "wrong" linear model is made in this chapter. To estimate the unknown parameter vector β^0 in a linear regression setting, a plethora of methods have been proposed starting from Gauss' least squares. In particular, we focus on the case where the data are possibly high-dimensional ($p > n$) or have additional structure that needs to be accounted for in the estimation procedure. In order to consistently estimate the unknown parameter vector β^0, it is often necessary to assume sparsity or one of its modifications depending on the specific problem. The (sharp) oracle inequalities are a theoretical instrument to measure the performance of an estimator for β^0.

1.2 Penalized Empirical Risk Minimization

From now on, we restrict the attention to parametric models. A commonly used class of methods to estimate the unknown parameters is *empirical risk minimization*. We assume that the distribution of the data $\{Z_i\}$ depends on a parameter $\beta \in \mathcal{C} \subseteq \mathbb{R}^p$. The loss function is then a function such that

$$\rho : \mathcal{Z} \times \mathcal{C} \to \mathbb{R}.$$

The "best" fit to the data is obtained by minimizing the empirical risk which is defined as:

$$R_n(\beta) = \frac{1}{n}\sum_{i=1}^{n} \rho(Z_i, \beta).$$

The population counterpart, the *risk*, is defined as

$$R(\beta) = \mathbb{E}R_n(\beta).$$

The quantity we are interested in computing is assumed to be

$$\beta^0 = \arg\min_{\beta\in\mathcal{C}} R(\beta).$$

In a high-dimensional setting, when the number of unknown parameters p exceeds the sample size n, we need to impose some additional structure in order to be able to compute a solution. This goal is usually achieved by adding a norm penalty $\Omega(\cdot) : \mathbb{R}^p \to \mathbb{R}_{\geq 0}$ to the empirical risk. In total one seeks to

$$\text{minimize } R_n(\beta) + \lambda\Omega(\beta), \tag{1}$$

for $\beta \in \mathcal{C}$, where $\lambda > 0$ is a tuning parameter. This optimization problem bears some intrinsic challenges. Depending on the choice of the loss, it might be nonconvex and/or nondifferentiable. From a computational point of view, choosing a norm as a penalty has the advantage that one might use (proximal) gradient descent algorithms. Often, the only hope is to obtain a stationary point of the optimization problem (1) instead of the minimizer. Stationary points are vectors that satisfy first-order *necessary* optimality conditions. A point $\tilde{\beta} \in \mathcal{C}$ is said to be a stationary point if for all $\beta \in \mathcal{C}$ we have

$$\left(\dot{R}_n(\tilde{\beta}) + \lambda\tilde{z}\right)^T (\beta - \tilde{\beta}) \geq 0, \tag{2}$$

where $\dot{R}_n(\tilde{\beta}) = \frac{\partial}{\partial\beta'} R_n(\beta')|_{\beta'=\tilde{\beta}}$ and $\tilde{z}$ is in the sub-differential $\partial\Omega(\tilde{\beta})$. If R_n is convex, the condition (2) is also *sufficient* for optimality of $\tilde{\beta}$.

The theoretical instrument we extend and/or newly derive for a variety of statistical estimation problems are *sharp oracle inequalities*. An oracle inequality compares the risk of an estimator $\hat{\beta}$ with the risk of an oracle. An oracle can be primarily any non-random vector $\beta^* \in \mathbb{R}^p$. A *sharp* oracle inequality is an inequality of the type

$$R(\hat{\beta}) \leq R(\beta^*) + \text{ estimation error.}$$

The sharpness is referred to the constant one in front of $R(\beta^*)$. Rewriting the above inequality, we see that

Table 1 General frameworks to derive sharp oracle inequalities

	Convex	Nonconvex
Differentiable	✓ Chapter 7 in [30]	✓ [10]
Nondifferentiable	✓ Section 2.1	?

$$R(\hat{\beta}) - R(\beta^0) \leq \underbrace{R(\beta^*) - R(\beta^0)}_{\text{approximation error}} + \text{estimation error.}$$

If an inequality of this type holds, it can be interpreted as follows: the approximation error does not have a large influence on the statistical performance of the estimator. In case of model misspecification, there will clearly be a component due to the approximation error that cannot be neglected.

In Chapter 7 of [30], a general framework for convex estimation problems is proposed for deriving sharp oracle inequalities. This general framework is extended to the case of nonconvex differentiable loss functions in [10]. In addition, in the present chapter, sharp oracle inequalities are shown to hold also for convex but nondifferentiable loss functions. There has been an increasing interest in methodologies leading to nonconvex optimization problems. We will therefore present applications of our theory to regression and classification-type problems as well as to principal component analysis.

A main novelty in this chapter is the sharp oracle inequality for minimizers of nondifferentiable convex loss functions as given in Sect. 2.1.

We summarize the types of loss functions to which the different frameworks can be applied in Table 1.

1.3 Conditions on the Risk

To guarantee a sufficient identifiability of the quantity of interest β^0, it is required that the risk is sufficiently curved/convex. For our results to hold, we make use of two very similar notions of strong convexity of the risk.

Condition 1 *Let G be an increasing strictly convex function $G : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ such that $G(0) = 0$. Let τ be a semi-norm on $\mathbb{R}^p$. For all $\beta_1, \beta_2 \in \mathcal{C}$ it holds that*

$$R(\beta_1) - R(\beta_2) - \dot{R}(\beta_2)^T(\beta_1 - \beta_2) \geq G(\tau(\beta_1 - \beta_2)).$$

The quantity on the left-hand side of the inequality is also called Bregman divergence. It measures the "distance" between any two points $\beta_1, \beta_2 \in \mathcal{C}$ in terms of the *vertical* blue line in Fig. 1. By requiring that this "distance", which is actually not a proper

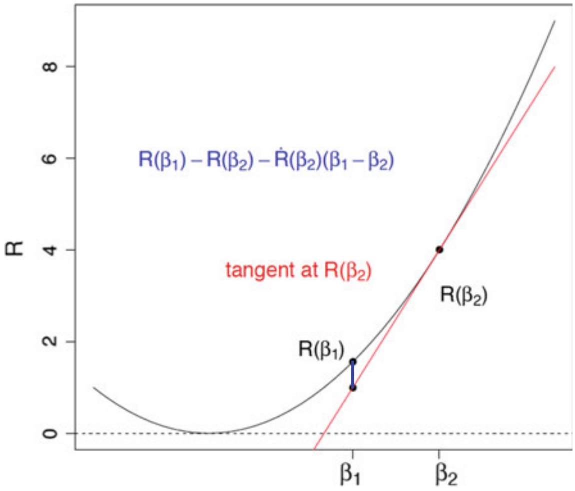


Fig. 1 The distance between β_1 and β_2 is expressed in terms of the vertical blue line. The requirement that this "distance" should always be nonnegative around β^0 translates to a requirement on the convexity of the risk

distance as it is not symmetric, is always nonnegative one automatically imposes a condition on the curvature of the risk.

A similar condition is needed for the case of nondifferentiable but convex loss functions. Despite the fact that the expectation "smooths" the nondifferentiable loss, we need to impose a condition on a difference quotient. Indeed, we have for a univariate real-valued differentiable function h that

$$\lim_{t\to 0}\frac{h((1-t)x_0+tx)-h(x_0)}{t}=\dot{h}(x_0).$$

As we need to consider differences of nondifferentiable averages (R_n) and to describe their concentration behavior around their expectation (R), we need a different notion of strong convexity given in the following condition.

Condition 2 *(G-convexity) For the same assumptions on G as in Condition 1 we say that G*-convexity *holds if* $\exists\ 0<t<1$ *such that*

$$(1-t)R(b)+tR(\beta)-tG\Big(\tau(\beta-b)\Big)\geq R\Big((1-t)b+t\beta\Big)$$

for all $b\in\mathcal{C}$.

1.4 Norm Penalties

As far as the penalty term is concerned, it is required to satisfy weak decomposability which might be seen as an inverse triangle inequality. Depending on the context, the notation β_S refers to the entries of the vector β whose indices are in S and otherwise zeros or only the entries of β whose indices are in S. It will be clear from the context if $\beta_S \in \mathbb{R}^p$ or $\beta_S \in \mathbb{R}^s$.

Definition 1 (Definition 4.1 in [29]) The norm $\Omega(\cdot)$ on $\mathbb{R}^p$ is said to be weakly decomposable if for some set $S \subseteq \{1, \dots, p\}$ with cardinality s there is another norm $\Omega^{S^c}(\beta_{S^c})$ on $\mathbb{R}^{p-s}$ so that

$$\Omega(\beta) \geq \Omega(\beta_S) + \Omega^{S^c}(\beta_{S^c}) =: \underline{\Omega}(\beta)$$

for all $\beta \in \mathbb{R}^p$. In particular $\underline{\Omega}(\cdot)$ is a norm on $\mathbb{R}^p$.

The ℓ_1-norm trivially satisfies the weak decomposability. Indeed, for all $\beta \in \mathbb{R}^p$ and all $S \subseteq \{1, \dots, p\}$ we have

$$\|\beta\|_1 := \sum_{j=1}^{p} |\beta_j| = \sum_{j \in S} |\beta_j| + \sum_{j \in S^c} |\beta_j|.$$

An additional concept that is needed to derive the sharp oracle inequalities is *effective sparsity*. The effective sparsity quantifies (in the linear model) the influence of the design (i.e., the distribution of the X_i's) on the estimation performance. It is a "comparison" of the chosen penalty with the semi-norm $\tau(\cdot)$ in Conditions 1 and 2 which is typically induced by the design. It also depends on the noise present in the estimation problem. Often, more noise leads to a larger effective sparsity as the sets S and S^c are more difficult to distinguish.

Definition 2 (Adapted from Definition 4.3 in [29]) The Ω-effective sparsity for $S \subseteq \{1, \dots, p\}$ is defined as

$$\Gamma(\tau, L, S) = \min \left\{ \tau(\beta_S - \beta_{S^c}) : \beta \in \mathbb{R}^p, \Omega(\beta_S) = 1, \Omega^{S^c}(\beta_{S^c}) \leq L \right\}^{-1}.$$

In the examples considered here, the bounds on the effective sparsity $\Gamma(\tau, L, S)$ will be independent of L. In this case, the effective sparsity can be interpreted also as a scaling factor that inflates the sparsity (e.g., the cardinality of the active set S^0) by taking into account the identifiability of β^0. The following example shows how to compute/upper bound the effective sparsity for the ℓ_1-norm and the semi-norm $\tau(\cdot) = \|\Sigma_X(\cdot)\|_2$, where $\Sigma_X \in \mathbb{R}^{p \times p}$ is chosen to be a positive definite covariance matrix as it will be the case in some of the applications.

Example 1 We denote the cardinality of S by $s := |S|$. By Hölder's inequality, we have

$$\begin{aligned} \|\beta_S\|_1 \leq \sqrt{s}\|\beta_S\|_2 &\leq \sqrt{s}\|\beta\|_2 \\ &\leq \sqrt{\frac{s}{\Lambda_{\min}(\Sigma_X)}} \|\Sigma_X \beta\|_2, \end{aligned}$$

where we interpret β_S as an element of $\mathbb{R}^p$. Therefore,

$$\frac{\|\beta_S\|_1}{\|\Sigma_X \beta\|_2} \leq \sqrt{\frac{s}{\Lambda_{\min}(\Sigma_X)}}.$$

We then have the following upper bound for the effective sparsity:

$$\Gamma(\tau, L, S) \leq \sqrt{\frac{s}{\Lambda_{\min}(\Sigma_X)}}$$

with $\tau(\cdot) = \|\Sigma_X(\cdot)\|_2$.

1.5 Related Literature

Sparsity oracle inequalities were derived in particular for the ℓ_1-penalized least squares estimator and its modifications. The works [4, 7, 8, 25] derive inequalities of this type. The book [6] is devoted to ℓ_1-norm penalized estimators ranging from estimation error bounds to support recovery and oracle inequalities. The book [15] gives a very precise description of the techniques used to derive oracle inequalities for *convex* penalized empirical risk minimization. The terminology *sharp oracle inequality* was coined in the context of penalized empirical risk minimization for matrix completion and ℓ_1-penalized regression in [16]. The ℓ_1-penalized framework is further extended to include *structured sparsity* (i.e., a norm penalty different from the ℓ_1-norm) [1, 2, 13, 22, 24]. This is further extended to square root loss in [27] so that to account for unknown error variances. In [30], general loss functions and more general norm penalties are considered. However, the treatment of the empirical processes is mostly done by a dual norm inequality which is too rough for loss functions whose derivatives depend on the parameter value.

As far as matrix regression models are concerned, sharp oracle inequalities have been first derived in [16] for an estimator involving a quadratic loss function with nuclear norm penalty. In [9], a sharp oracle inequality for the Huber loss with nuclear norm penalty has been derived.

The interest in the (statistical) properties of stationary points has recently increased due to the plethora of methods used in subjects such as deep learning and neural networks. As a matter of fact, it is not possible to compute the *global* minima of the optimization problems (see for example [11]). From a statistical point of view, the properties of stationary points of regularized empirical risk minimization problems were studied in [19]. For the linear model with errors in variables, an estimator is obtained by replacing the sample covariance matrix with an unbiased estimator of the population version an estimator is proposed. This estimator is nonconvex in a high-dimensional ($p > n$) setting. A gradient descent-type algorithm is shown to converge to first-order optimal points. Upper bounds on the statistical performance of the stationary points are derived. The bounds on the estimation error depend on a term stemming from the algorithm and on the statistical estimation error.

A general framework to derive purely statistical error bounds for stationary points of penalized M-estimators is given in [20]. The notion of restricted strong convexity allows one to derive near-optimal statistical rates. The general theory can be applied also to nonconvex penalties. Nonconvex penalties are important as they might be used to diminish for example the bias of the Lasso estimates. In [21], the role of nonconvex penalties in support recovery is further examined. In particular, it is shown that stationary points stemming from estimators penalized with appropriate nonconvex functions require less assumptions than the Lasso to succeed with support recovery.

The work [18] further examines nonconvex robust loss functions. It is described in detail that some properties of nonconvex robust loss functions indeed outperform the convex losses such as the Huber loss. The statistical properties are shown to hold in a neighborhood of the target vector. To justify this assumption, a two-stage procedure based on an initial convex estimator is proposed.

In [23], the "landscape" of nonconvex M-estimators is considered. This framework requires different assumptions on the distribution of the observations than in the previously cited works on stationary points.

Finally, [10] provides a novel framework to derive *sharp* oracle inequalities for stationary points. It extends the estimation error rates obtained in the previously cited papers on the statistical properties of stationary points.

1.6 Organization

In Sect. 2, we give deterministic versions of the sharp oracle inequalities. In Sect. 3, the general deterministic theorems are applied to the specific examples from regression, classification, and dimension reduction. The following main ingredients are needed to obtain the sharp oracle inequalities:

(i) The convexity of the set $\mathcal{C}$ on which the optimization problem is solved.
(ii) The strong convexity of the risk R on $\mathcal{C}$ or alternatively the G-convexity of R.
(iii) The weak decomposability of the penalty.
(iv) The effective sparsity.
(v) A uniform bound on the random part of the estimation problem.

In the examples, the properties (i)–(v) will be verified.

2 Sharp Oracle Inequalities

For the sake of a clearer description, we first discuss purely deterministic theorems. The random part then needs to be accounted for in the specific applications. In this section, the random part is assumed to be bounded by what we call the "noise level"

λ_ε. The subscript ε does not necessarily mean that the quantity depends on (the distribution of) the (additive) noise. It should rather be seen as a general dependence on "some" *random* noise in the statistical estimation problem. In the applications of the general framework, the noise level will be chosen depending on the distributional assumptions of the data and the errors.

From now on, we use the notation $S^* = \{j \subseteq \{1, \dots, p\} : \beta_j^* \neq 0\}$ and $s^* = |S^*|$ to denote the support set of the vector β^* and its respective cardinality s^*.

2.1 Convex and Nondifferentiable

In this section, we describe a deterministic sharp oracle inequality for nondifferentiable convex loss functions.

Theorem 1 *Suppose that $R_n(\cdot)$ is convex and let $\hat{\beta}$ be the minimizer of (1). Assume G-convexity (Condition 2) and let H be the convex conjugate of G. Let $\gamma_n \geq 0$ and assume there exists $\lambda_\varepsilon > 0$ such that*

$$\lambda_\varepsilon \geq \frac{\left|\left[R_n - R\right]\left((1-t)\hat{\beta} + t\beta^*\right) - \left[R_n - R\right]\hat{\beta}\right|}{t(\underline{\Omega}(\hat{\beta} - \beta^*) + \gamma_n)}. \tag{3}$$

Then for $\lambda > \lambda_\varepsilon$ and defining for some $0 \leq \delta < 1$

$$\underline{\lambda} = \lambda - \lambda_\varepsilon, \quad \overline{\lambda} = \lambda + \lambda_\varepsilon + \delta\underline{\lambda}, \quad L = \frac{\overline{\lambda}}{(1-\delta)\underline{\lambda}}$$

we have

$$\delta\underline{\lambda}\underline{\Omega}(\hat{\beta} - \beta^*) + R(\hat{\beta}) \leq R(\beta^*) + H\left(\overline{\lambda}\Gamma(\tau, L, S^*)\right) + 2\lambda\Omega(\beta^*_{S^{*c}}) + \lambda_*,$$

where $\lambda_* = \lambda_\varepsilon \gamma_n$.

Due to the convexity of R_n, there is only one stationary point $\hat{\beta}$ which is also the global minimizer of the objective function (1). The proof of Theorem 1 can be found in the appendix. Assumption (3) ensures that the difference between averages and expectations in the estimation problem is bounded. To be able to measure the estimation error also in the $\underline{\Omega}(\cdot)$ norm, the parameter $0 \leq \delta < 1$ is introduced.

2.2 Nonconvex and Differentiable

The following theorem states the deterministic sharp oracle inequality for nonconvex differentiable loss functions.

Theorem 2 (Theorem 2.1 in [10]) *Let $\tilde{\beta}$ be a stationary point of (1). Suppose that Condition 1 is satisfied. Let H be the convex conjugate of G. Let for $0 \leq \gamma < 1$, $\lambda_* \geq 0$, $\lambda_\varepsilon > 0$ and for all $\beta' \in \mathcal{C}$*

$$\left|\left(\dot{R}_n(\beta') - \dot{R}(\beta')\right)^T (\beta^* - \beta')\right| \leq \lambda_\varepsilon \underline{\Omega}(\beta^* - \beta') + \gamma G(\tau(\beta^* - \beta')) + \lambda_* \tag{4}$$

and $\lambda > \lambda_\varepsilon$. For some $0 \leq \delta < 1$ define

$$\underline{\lambda} = \lambda - \lambda_\varepsilon, \quad \overline{\lambda} = \lambda + \lambda_\varepsilon + \delta\underline{\lambda}, \quad L = \frac{\overline{\lambda}}{(1-\delta)\underline{\lambda}}.$$

Then we have

$$\begin{aligned} \delta\underline{\lambda\Omega}(\tilde{\beta} - \beta^*) + &R(\tilde{\beta}) \\ &\leq R(\beta^*) + (1-\gamma)H\left(\frac{\overline{\lambda}\Gamma(\tau, L, S^*)}{(1-\gamma)}\right) + 2\lambda\Omega(\beta^*_{S^{*c}}) + \lambda_*. \end{aligned} \tag{5}$$

Remark 1 Theorem 2 applies to *any* stationary point of the objective function. We make use of the definition of stationary point as given in [3] and further used in [20] in a statistical context. This definition also comprises local maxima. Any point $\tilde{\beta}$ satisfying a sharp oracle inequality is shown to have a prediction performance on new unseen "testing" data that is almost as good as the prediction performance $R(\cdot)$ of an oracle. The oracle might be chosen as a vector that minimizes the upper bound in inequality (5).

Remark 2 The quantity λ_ε is to be seen as an upper bound on the stochastic part of an estimation problem. In the linear regression setting with predictors $X \in \mathbb{R}^{n \times p}$ and independent errors $\varepsilon \in \mathbb{R}^n$ λ_ε is an upper bound on $\|X^T\varepsilon\|_\infty/n$. For this reason, it is often named "noise level".

2.3 Asymptotic Interpretation

An asymptotic interpretation of Theorems 1 and 2 under an appropriate scaling of the sample size, dimension, sparsity, and (non-)sparsity of the problem can be read as

$$R(\tilde{\beta}) = R(\beta^*) + O\left(H\left(\frac{\overline{\lambda}\Gamma(\tau, L, S^*)}{(1-\gamma)}\right)\right).$$

This means that the risk/performance of the estimator is almost as good as the performance of an oracle β^*. It may be chosen to optimally trade off the approximation and estimation errors. The oracle may be chosen as the vector β^* that minimizes the upper bounds of the sharp oracle inequality. As a consequence, the points $\tilde{\beta}$ satisfying the sharp oracle inequalities will have a risk/prediction performance that is comparable to the best approximation within the model. The first term is a constant, whereas the second term is the estimation error in the specific applications that decreases with increasing sample size.

3 Applications

The nonconvexity of the optimization problems obviously appears in the second derivatives of the (empirical) risk. In the following applications, one therefore needs to more closely consider the properties of the Hessian matrices. In particular, one needs to verify either Conditions 1 or 2. As far as the empirical part is concerned, a considerable amount of work goes into showing that the empirical processes (3) and (4) are bounded with high probability. In particular, this involves bounding (sparse) random quadratic forms. For each application, we will go through a standard recipe in order to establish the conditions and assumptions needed in the general frameworks.

3.1 Regression

The most classical field of application of (sharp) oracle inequalities is regression. Many of the examples are therefore related to it.

3.1.1 Sparse Corrected Linear Regression

We consider the linear model

$$Y = X\beta^0 + \varepsilon,$$

where for all $i = 1, \ldots, n$ it is assumed that $\varepsilon_i \in \mathbb{R}$ are i.i.d. sub-Gaussian. The matrix $X \in \mathbb{R}^{n\times p}$ is assumed to have i.i.d. sub-Gaussian rows $X_i \in \mathbb{R}^p$ with positive definite covariance matrix Σ_X for all $i = 1, \ldots, n$ independent of ε. The matrix $W \in \mathbb{R}^{n\times p}$ is assumed to have i.i.d. sub-Gaussian rows $W_i \in \mathbb{R}^p$ for all $i = 1, \ldots, n$, to have a known positive definite covariance matrix Σ_W and to be independent of X and ε. We define

$$Z = X + W.$$

It is assumed that the pair $(Y, Z) \in \mathbb{R}^n \times \mathbb{R}^{n\times p}$ is observed and that $(\varepsilon, W, X) \in \mathbb{R}^n \times \mathbb{R}^{n\times p} \times \mathbb{R}^{n\times p}$ is unobserved.

An estimator for β^0 is then given by

$$\hat{\beta} = \underset{\beta\in\mathbb{R}^p:\|\beta\|_1\le Q}{\arg\min} \underbrace{\frac{1}{2}\beta^T\left(\frac{Z^TZ}{n} - \Sigma_W\right)\beta - \frac{Y^TZ}{n}\beta}_{=R_n(\beta)} + \lambda\|\beta\|_1, \tag{6}$$

where $\lambda > 0$ and $Q > 0$ are tuning parameters. Note that this estimator very much resembles the convex estimator Lasso (see [28]): The matrix $(Z^TZ/n - \Sigma_W)$ is an unbiased estimator for Σ_X and Y^TZ/n is an unbiased estimator for $\beta^{0^T}\Sigma_X$. The second derivative of the empirical risk is given by

$$\ddot{R}_n(\beta) = \frac{Z^TZ}{n} - \Sigma_W.$$

The sample version is therefore nonconvex for $p > n$. We now verify the conditions needed to apply Theorem 2:

(i) The set $\mathcal{C} = \{\beta \in \mathbb{R}^p : \|\beta\|_1 \le Q\}$ is convex.

(ii) Condition 1 is satisfied with $G(\cdot) = (\cdot)^2$ and $\tau(\cdot) = \left\|\Sigma_X^{1/2}(\cdot)\right\|_2$. For all $\beta_1, \beta_2 \in \mathbb{R}^p$ we have

$$\begin{aligned} R(\beta_1) - R(\beta_2) - \dot{R}(\beta_2)^T(\beta_1 - \beta_2) &= (\beta_1 - \beta_2)^T\Sigma_X(\beta_1 - \beta_2) \\ &= \left\|\Sigma_X^{1/2}(\beta_1 - \beta_2)\right\|_2^2 \\ &= G(\tau(\beta_1 - \beta_2)). \end{aligned}$$

(iii) The penalty is the ℓ_1−norm: $\Omega(\cdot) = \|\cdot\|_1$.

(iv) The effective sparsity can be bounded as follows

$$\Gamma(\|\Sigma^{1/2}(\cdot)\|_2, L, S^*) \le \sqrt{\frac{s^*}{\Lambda_{\min}(\Sigma_X)}}.$$

(v) With probability at least $1 - 5\exp(-\log(2p))$ and with $\lambda_\varepsilon = C\sqrt{\frac{\log p}{n}}$ and assuming $n \ge c\log p$, we have

$$\left|\left(\dot{R}_n(\beta') - \dot{R}(\beta')\right)^T(\beta^* - \beta')\right| \le \lambda_\varepsilon\|\beta^* - \beta'\|_1 + \gamma\|\Sigma_X^{1/2}(\beta^* - \beta')\|_2^2,$$

for all $\beta' \in \mathcal{C}$ as shown in Lemma 16 in [10].

Having verified all the conditions of Theorem 2, we have the following corollary for all stationary points of the optimization problem (6).

Corollary 1 (Corollary 3.1 in [10]) *Let $\tilde{\beta}$ be a stationary point of (6). We then have with probability at least* $1 - 5\exp(-\log(2p))$

$$\delta\underline{\lambda}\|\tilde{\beta} - \beta^*\|_1 + R(\tilde{\beta}) \leq R(\beta^*) + \frac{\overline{\lambda}^2 s^*}{4\Lambda_{\min}(\Sigma_X)(1-\gamma)} + 2\lambda\|\beta^*_{S^{*c}}\|_1.$$

3.1.2 Sparse Robust Regression

The linear regression model

$$Y = X\beta^0 + \varepsilon$$

is considered with $\beta^0 \in \mathbb{R}^p$, $X = (X_1, \ldots, X_n)^T \in \mathbb{R}^{n\times p}$ and $X_i \in \mathbb{R}^p$ i.i.d. sub-Gaussian with positive definite covariance matrix Σ_X and noise vector $\varepsilon \in \mathbb{R}^n$ independent of X and possibly heavy-tailed. An estimator for β^0 restricted on the set $\mathcal{B} = \left\{\beta \in \mathbb{R}^p : \|\beta - \beta^0\|_2 \leq \eta\right\}$ for some constant $\eta > 0$ is then given by

$$\hat{\beta} = \arg\min_{\beta\in\mathcal{B}} \frac{1}{n}\sum_{i=1}^n \rho(Y_i - X_i\beta) + \lambda\|\beta\|_1, \tag{7}$$

where $\lambda > 0$ is a tuning parameter.

Convex nondifferentiable loss function

We consider

$$R_n(b) := \frac{1}{n}\sum_{i=1}^n \rho(Y_i - X_i b),$$

with $\rho(z) := |z|$, $z \in \mathbb{R}$ and a sparsity-inducing estimator

$$\hat{\beta} = \arg\min_{\beta\in\mathcal{B}} R_n(\beta) + \lambda\|\beta\|_1. \tag{8}$$

Remark 3 Because $R_n(\cdot)$ is convex one only needs conditions in a convex neighborhood $\mathcal{B}$ of β^0 (i.e., *local* conditions). We do not detail this to avoid digressions.

The distributional assumptions in this case are as follows:

(1) For all $b \in \mathcal{B}$ and all $i = 1, \ldots, n$, it holds that $|X_i(b - \beta^0)| \leq K$ for some constant $K > 0$.

(2) The errors $\epsilon_1, \ldots, \epsilon_n$ are i.i.d. with median zero and a positive density f_ϵ near zero: for some positive constants K and C

$$\underline{C}^2 \leq f_\epsilon(z) \leq \bar{C}^2,\ \forall\, |z| \leq K.$$

To apply Theorem 1, we need to verify the properties i) – v):

(i) The set $\mathcal{C} = \mathcal{B}$ is convex.
(ii) The G-convexity of R holds.

Lemma 1 *Let $[a, b]$ be an interval in $\mathbb{R}$ and $g : [a, b] \rightarrow \mathbb{R}$ be a function with, for some positive constants $\bar{C}^2$ and $\underline{C}^2$,*

$$1/\underline{C}^2 \leq \ddot{g}(v) \leq \bar{C}^2, \ \forall\, v \in [a, b].$$

Then for $t \leq 1/(2\bar{C}^2\underline{C}^2)$ it holds that

$$g((1-t)u + tv) \leq (1-t)g(u) + tg(v) - \tfrac{1}{4}t(v-u)^2/\underline{C}^2.$$

A proof of Lemma 1 is given in the appendix.

Then for b and $\beta \in \mathbb{R}^p$ and for

$$r_i(b) := \mathbb{E}\left(|Y_i - X_i b| \,\middle|\, X_i \right),$$

we have with $t \leq 1/(2\bar{C}^2\underline{C}^2)$ the inequality

$$r_i((1-t)b + t\beta) \leq (1-t)r_i(b) + tr_i(\beta) - \tfrac{1}{4}t(X_i(b-\beta))^2/\underline{C}^2.$$

Therefore, for b and β in $\mathcal{B}$

$$R((1-t)b + t\beta) \leq (1-t)R(b) + tR(\beta) - \tfrac{1}{4}t\|\Sigma_X^{1/2}(b-\beta)\|_2^2/\underline{C}^2.$$

Thus, G-convexity holds with $\tau(b) = \|\Sigma_X^{1/2} b\|_2$ and $G(u) = \frac{1}{4}u^2/\underline{C}^2$. Moreover, the constant t can be chosen as $t = 1/(2\bar{C}^2\underline{C}^2)$.

(iii) The penalty is the ℓ_1-norm: $\Omega(\cdot) = \|\cdot\|_1$.
(iv) The effective sparsity can be bounded as follows

$$\Gamma(\|\Sigma_X^{1/2}(\cdot)\|_2, L, S^*) \leq \sqrt{\frac{s^*}{\Lambda_{\min}(\Sigma_X)}}.$$

(v) Choosing the noise level for some absolute constant $C > 0$ as $\lambda_\varepsilon = C\sqrt{\frac{\log p}{n}}$, we have for all $\beta \in \mathcal{B}$ with probability at least $1 - c' \exp(-c \log p)$ that

$$\lambda_\varepsilon \geq \frac{\left| \left[R_n - R \right] \left((1-t)\hat{\beta} + t\beta \right) - \left[R_n - R \right] \hat{\beta} \right|}{t(\|\hat{\beta} - \beta\|_1 + \gamma_n)}.$$

Remark 4 Condition (v) can be checked using the Lipschitz property of the absolute value loss. This allows one to apply the contraction inequality (see for example Chap. 4 in [17], the version that is referred to here is given in Theorem 16.2 in [30]).

Hence, Theorem 1 can be applied to the global minimum $\hat{\beta}$.

Corollary 2 *Let $\hat{\beta}$ be the solution of the optimization problem (8). We then have with probability at least* $1 - c' \exp(-c \log p)$

$$\delta\underline{\lambda}\|\hat{\beta} - \beta^*\|_1 + R(\hat{\beta}) \leq R(\beta^*) + \frac{\underline{C}^2 \overline{\lambda}^2 s^*}{\Lambda_{\min}(\Sigma_X)} + 2\lambda\|\beta^*_{S^{*c}}\|_1 + \lambda_*.$$

Nonconvex differentiable loss functions

In this section, we define $\mathcal{C} = \mathcal{B} \cap \{\beta \in \mathbb{R}^p : \|\beta\|_1 \leq Q\}$. The estimator is then given by optimizing the following penalized empirical risk:

$$\hat{\beta} = \arg\min_{\beta \in \mathcal{C}} \frac{1}{n} \sum_{i=1}^{n} \rho(Y_i - X_i\beta) + \lambda\|\beta\|_1. \tag{9}$$

Here, in contrast to the convex case, the restriction to the neighborhood $\mathcal{B}$ corresponds to assuming that a good initialization is available. The initial point is assumed to be sufficiently close to β^0. In this case, one might, for instance, choose the Huber loss as done, e.g., in [18].

The assumptions on the loss function are as follows:

1. The loss function $\rho : \mathbb{R} \to \mathbb{R}$ is at least twice continuously differentiable.
2. The loss function is Lipschitz continuous: there exists a constant $\kappa_1 > 0$ such that for all $x \in \mathbb{R}$
$$|\dot{\rho}(x)| \leq \kappa_1.$$
3. The first derivative of the loss function is Lipschitz continuous: there exists a constant $\kappa_2 > 0$ such that for all $x \in \mathbb{R}$
$$|\ddot{\rho}(x)| \leq \kappa_2.$$

We now verify the assumptions needed for Theorem 2:

(i) The set $\mathcal{C} = \mathcal{B} \cap \{\beta \in \mathbb{R}^p : \|\beta\|_1 \leq Q\}$ is convex.

(ii) For all $\beta_1, \beta_2 \in \mathcal{C}$ we have for some constant $C > 0$ that depends on the distribution of the noise

$$R(\beta_1) - R(\beta_2) - \dot{R}(\beta_2)^T(\beta_1 - \beta_2) \geq \underbrace{C\|\Sigma_X^{1/2}(\beta_1 - \beta_2)\|_2^2}_{=G(\tau(\beta_1-\beta_2))}.$$

(iii) The penalty is given by $\Omega(\cdot) = \|\cdot\|_1$.

(iv) The effective sparsity can be bounded as follows

$$\Gamma(\|\Sigma_X^{1/2}(\cdot)\|_2, L, S^*) \leq \sqrt{\frac{s^*}{\Lambda_{\min}(\Sigma_X)}}.$$

(v) With probability at least $1 - c\exp(-c'\log p)$, a noise level $\lambda_\varepsilon = C\sqrt{\frac{\log p}{n}}$, and a sufficiently large sample size $n \geq cs^*\log p$, we have

$$\left(\dot{R}_n(\beta') - \dot{R}(\beta')\right)^T(\beta^* - \beta') \leq \lambda_\varepsilon\|\beta^* - \beta'\|_1 + \gamma G(\tau(\beta^* - \beta')),$$

for all $\beta' \in \mathcal{C}$ as done in Lemma 31 of [10].

Corollary 3 (Corollary 3.3 in [10]) *Let $\tilde{\beta}$ be a stationary point of (9). Then we have with probability at least* $1 - c\exp(-c'\log p)$ *that*

$$\delta\underline{\lambda}\|\tilde{\beta} - \beta^*\|_1 + R(\tilde{\beta}) \leq R(\beta^*) + \frac{\overline{\lambda}^2 s^*}{C\Lambda_{\min}(\Sigma_X)(1-\gamma)} + 2\lambda\|\beta^*_{S^{*c}}\|_1.$$

3.1.3 Robust SLOPE

We propose a new estimator named robust Sorted ℓ_1-penalized estimator (SLOPE) that is inspired by the estimator SLOPE proposed in [5]. The aim of this example is to demonstrate how the general framework can be applied to penalties different from the vector ℓ_1-norm. With $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_p > 0$, the sorted ℓ_1-norm is defined as

$$J_\mu(\beta) = \sum_{j=1}^{p} \mu_j |\beta|_{(j)},$$

where $|\beta|_{(1)} \geq \cdots \geq |\beta|_{(p)}$ are the ordered absolute values of the entries of the vector β. In [27], it is shown that the sorted ℓ_1-norm is indeed weakly decomposable: for some $S \subseteq \{1, \ldots, p\}$ and all $\beta \in \mathbb{R}^p$, we have

$$J_\mu(\beta) \geq J_\mu(\beta_S) + \sum_{l=1}^{r} \mu_{p-r+l}|\beta|_{(l,S^c)} =: \underline{\Omega}(\beta),$$

where $r = p - s$. The robust SLOPE estimator is then given by

$$\hat{\beta} = \underset{\beta \in \mathcal{B}: \|\beta\|_1 \leq Q}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} \rho(Y_i - X_i\beta) + \lambda J_\mu(\beta), \tag{10}$$

where $\lambda > 0$ and $Q > 0$ are tuning parameters. As far as the loss function is concerned, we require the same conditions as in Sect. 3.1.2. The only part that changes is the part on the penalty. The following holds:

(iii) The penalty is given by the norm $J_\mu(\beta)$.
(iv) The effective sparsity can be bounded as follows

$$\Gamma(\|\Sigma_X^{1/2}(\cdot)\|_2, L, S^*) \leq \mu_1 \sqrt{\frac{s^*}{\Lambda_{\min}(\Sigma_X)}}.$$

A sharp oracle inequality for the stationary points of (10) is given in the following corollary.

Corollary 4 (Corollary 3.5 in [10]) *Let* $\tilde{\beta}$ *be a stationary point of (10). Then we have with probability at least* $1 - c\exp(-c' \log p)$ *that*

$$\delta \underline{\lambda \Omega}(\tilde{\beta} - \beta^*) + R(\tilde{\beta}) \leq R(\beta^*) + \frac{\overline{\lambda}^2 s^*}{C \Lambda_{\min}(\Sigma_X)(1-\gamma)} + 2\lambda J_\mu(\beta^*_{S^{*c}}).$$

3.2 Classification

In classification, the aim is to assign a label to a given set of observations. For simplicity, we code the labels Y_i to take values in $\{0, 1\}$. The conditional probabilities can be modeled as

$$\mathbb{P}(Y_i = 1 | X_i = x_i) = \frac{\exp(x_i\beta^0)}{1 + \exp(x_i\beta^0)} =: \sigma(x_i\beta^0).$$

For some $\eta > 0$ define $\mathcal{B} = \left\{\beta' \in \mathbb{R}^p : \|\beta' - \beta^0\| \leq \eta\right\}$. We consider the following sparsity-inducing estimator for β^0:

$$\hat{\beta} = \underset{\beta \in \mathcal{B}}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} (Y_i - \sigma(x_i\beta))^2 + \lambda\|\beta\|_1, \tag{11}$$

where $\lambda > 0$ is a tuning parameter. The assumptions on the distribution of the features X_i are as follows:

1. The features X_i are assumed to be i.i.d. sub-Gaussian with constant $C_X > 0$ and positive definite covariance matrix Σ_X for all $i = 1, \dots, n$.
2. It is assumed that for all $i = 1, \dots, n$ and for the oracle $\beta^* \in \mathcal{B} : |X_i \beta^*| \leq K$ almost surely, where K is some positive constant.

We have the following properties of the estimation problem:

(i) The set $\mathcal{C} = \mathcal{B}$ is convex.

(ii) If $\min_{s \in [-K,K]} \sigma'(s) > C_X^3 \eta \Lambda_{\min}(\Sigma_X)^{-1}$ we have for all $\beta_1, \beta_2 \in \mathcal{C}$

$$R(\beta_1) - R(\beta_2) - \dot{R}(\beta_2)^T(\beta_1 - \beta_2) \geq C(K, \eta, \Sigma_X)\|\beta_1 - \beta_2\|_2^2 =: G(\tau(\beta_1 - \beta_2)).$$

(iii) The penalty is the ℓ_1-norm: $\Omega(\cdot) = \|\cdot\|_1$.

(iv) The effective sparsity can be bounded as follows

$$\Gamma(\|\cdot\|_2, L, S^*) \leq \sqrt{s^*}.$$

(v) With probability at least $1 - c\log_2(p)\exp(-\log p)$ and choosing $\lambda_\varepsilon = C\sqrt{\frac{\log p}{n}}$ we have for all $\beta' \in \mathcal{B}$

$$\left|\left(\dot{R}_n(\beta') - \dot{R}(\beta')\right)^T(\beta^* - \beta')\right| \leq \lambda_\varepsilon \|\beta' - \beta^*\|_1 + \frac{C' \log p}{n},$$

which is shown to hold in Lemma 37 in [10].

Corollary 5 (Corollary 3.4 in [10]) *Let $\tilde{\beta}$ be a stationary point of (11). Then we have with probability at least* $1 - c\log_2(p)\exp(-\log p)$ *that*

$$\delta\underline{\lambda}\|\tilde{\beta} - \beta^*\|_1 + R(\tilde{\beta}) \leq R(\beta^*) + \frac{\overline{\lambda}^2 s^*}{C\Lambda_{\min}(\Sigma_X)^2} + 2\lambda\|\beta_{S^{*c}}\|_1 + \frac{C' \log p}{n}.$$

3.3 Dimension Reduction

In this section, we demonstrate that the general framework can be applied also to an estimation problem from unsupervised learning. Dimension reduction is important to enhance interpretability and to summarize the information contained in a given data set. The probably most prominent method is Principal Component Analysis (PCA). In the sequel, it is demonstrated how to case PCA as a (penalized) empirical risk minimization problem and how to apply the general framework to derive a sharp oracle inequality.

3.3.1 Sparse Principal Component Analysis

Principal Component Analysis (PCA) is used to represent data in a concise way. To obtain an estimate of the loadings of the first Principal Component (PC), one maximizes the sample variance subject to a constraint on the length. The subsequent principal components are computed similarly by imposing additional orthogonality constraints. More precisely, given a data matrix $X \in \mathbb{R}^{n \times p}$ with i.i.d. rows $X_1, \ldots, X_n \in \mathbb{R}^p$. The sample covariance matrix is then given by $\hat{\Sigma} = X^T X/n$. For convenience, we assume that the observations stem from a distribution with mean zero and positive definite covariance matrix $\Sigma_X = \mathbb{E}\hat{\Sigma}$. To estimate the first PC, one solves the following optimization problem with respect to $\beta \in \mathbb{R}^p$:

$$\text{maximize } \beta^T \hat{\Sigma} \beta \text{ subject to } \|\beta\|_2 = 1.$$

One can also think of this optimization problem as of computing the best rank one approximation in squared Frobenius norm of the sample covariance matrix $\hat{\Sigma}$:

$$\text{minimize } \frac{1}{4} \| \hat{\Sigma} - \beta\beta^T \|_F^2.$$

These two approaches are equivalent up to a normalizing constant. From the latter representation of the optimization problem, it can be seen that it is nonconvex. Despite the nonconvexity and when $n > p$, PCA produces a consistent estimator: the eigenvector corresponding to the largest eigenvalue of $\hat{\Sigma}$. When the dimension of the parameter vector exceeds the sample size ($p \geq n$), it has been shown in [14] that the first principal component cannot be consistently estimated. Assuming sparsity of the first principal component enables to consistently estimate it by means of a sparsity-inducing estimator. The estimator under study is given by

$$\begin{aligned} \hat{\beta} &= \arg\min_{\beta \in \mathcal{C}} \frac{1}{4} \| \hat{\Sigma} - \beta\beta^T \|_F^2 + \lambda \|\beta\|_1 \\ &= \arg\min_{\beta \in \mathcal{C}} R_n(\beta) + \lambda \|\beta\|_1, \end{aligned} \tag{12}$$

where $\lambda > 0$ is a tuning parameter. The optimization problem (12) is nonconvex and it cannot be solved explicitly. Gradient descent-type algorithms applied to the optimization problem (12) are guaranteed to "output" a stationary point in the sense of equation (2).

This estimation problem exhibits a twofold nonconvexity: the *population version* as well as the empirical version are nonconvex. In order to match our framework, it is therefore necessary to first decide which β^0 should be estimated. In PCA, the multiple minima are the same up to sign changes. In Fig. 2, it is demonstrated by means of a toy

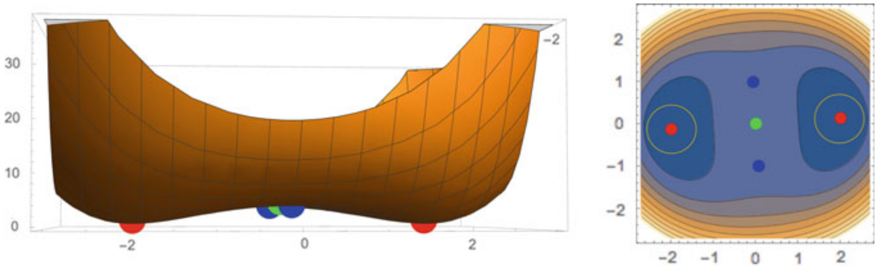


Fig. 2 Left: Risk function using a 2 × 2 positive definite covariance matrix. The red points represent the first principal component vector (up to a sign flip). Right: Contour plot of the same risk function as in the left figure. The circles around the red points represent the neighborhoods $\mathcal{B}$. The remaining dots represent other stationary points

example as done in [12] how the risk function looks like. To guarantee a sufficiently large curvature around the optima, it is necessary to impose certain conditions on the singular values of the population covariance matrix.

The eigendecomposition of Σ_X shall be given by

$$\Sigma_X = Q\Phi^2 Q^T,$$

where $\Phi = \text{diag}(\phi_1, \ldots, \phi_p)$, $\phi_{\max} = \phi_1 \geq \cdots \geq \phi_p > 0$ and $Q^T Q = QQ^T = I_{p\times p}$. In order to guarantee a sufficient curvature around β^0, the difference between a so-called *spikiness condition* needs to be imposed. It says that the largest and second largest singular values of the population covariance matrix must be sufficiently well separated: let $\xi > 0$ be the "eigengap". It is assumed that

$$\phi_{\max} \geq \phi_j + \xi, \text{ for all } j \neq 1.$$

The following lemma shows that for $\eta > 0$ the risk is convex on the set $\mathcal{B} := \left\{\beta \in \mathbb{R}^p : \|\beta - \beta^0\|_2 \leq \eta\right\}$.

Lemma 2 (Lemma 12.7 in [30]) *Assume that $\xi > 3\eta$. Then we have for all $\beta \in \mathcal{B}$*

$$\Lambda_{\min}(\ddot{R}(\beta)) \geq 2\phi_{\max}(\xi - 3\eta),$$

where $\Lambda_{\min}(\ddot{R}(\beta))$ is the smallest eigenvalue of $\ddot{R}(\beta)$.

The conditions to apply Theorem 2 are

(i) The set $\mathcal{C} := \left\{\beta \in \mathbb{R}^p : \|\beta - \beta^0\|_2 \leq \eta\right\} \cap \{\beta \in \mathbb{R}^p : \|\beta\|_1 \leq Q\}$ is convex.
(ii) Condition 1 is satisfied assuming that $\xi > 3\eta$. We have for all $\beta_1, \beta_2 \in \mathcal{C}$

$$R(\beta_1) - R(\beta_2) - \dot{R}(\beta_2)^T(\beta_1 - \beta_2) \geq \underbrace{2(\xi - 3\eta)\phi_{\max}\|\beta_1 - \beta_2\|_2^2}_{=G(\tau(\beta_1 - \beta_2))}.$$

(iii) The penalty is the ℓ_1-norm: $\Omega(\cdot) = \|\cdot\|_1$.

(iv) The effective sparsity can be bounded as follows

$$\Gamma(\|\cdot\|_2, L, S^*) \leq \sqrt{s^*}.$$

(v) With probability at least $1 - 2\exp(-\log(2p))$, with $\lambda_\varepsilon = C\sqrt{\frac{\log p}{n}}$ and assuming a sufficiently large sample size $n \geq c \log p$ we have for all $\beta' \in \mathcal{C}$ that

$$\left|\left(\dot{R}_n(\beta') - \dot{R}(\beta')\right)^T (\beta^* - \beta')\right| \leq \lambda_\varepsilon\|\beta' - \beta^*\|_1 + \gamma G(\tau(\beta' - \beta^*))$$

as shown in Lemma 25 in [10].

Corollary 6 (Corollary 3.2 in [10]) *Let $\tilde{\beta} \in \mathcal{C}$ be a stationary point of (12). Then we have with probability at least* $1 - 2\exp(-\log(2p))$

$$\delta\underline{\lambda}\|\tilde{\beta} - \beta^*\|_1 + R(\tilde{\beta}) \leq R(\beta^*) + \frac{\overline{\lambda}^2 s^*}{8(\xi - 3\eta)\phi_{\max}(1 - \gamma)} + 2\lambda\|\beta^*_{S^{*c}}\|_1.$$

4 Discussion

We have described the concept of (sharp) oracle inequality by means of the linear model and summarized two extensions of a general framework to derive this type of inequalities. The first extension is for nondifferentiable convex loss functions. There, a slightly differently stated strong convexity requirement on the risk function makes it possible to extend the sharp oracle inequalities also to this case. The second extension is concerned with nonconvex loss functions. As a matter of fact, the minimum of a nonconvex loss function is almost impossible to compute via the known (proximal) gradient descent-type algorithms. On the other hand, these algorithms are guaranteed to stop at points satisfying first-order necessary optimality conditions (i.e., stationary points). For these points of nonconvex loss functions, we have demonstrated how to apply a general theorem to some specific estimation problems. The examples include a convex and nondifferentiable loss function as well as nonconvex and differentiable loss functions from regression, classification, and dimension reduction.

5 Appendix

Proof of Theorem 1

Let $\hat{\beta}_t := (1-t)\hat{\beta} + t\beta^*$. It holds that

$$\begin{aligned} R_n(\hat{\beta}) + \lambda\Omega(\hat{\beta}) &\le R_n(\hat{\beta}_t) + \lambda\Omega(\hat{\beta}_t) \\ &\le R_n(\hat{\beta}_t) + (1-t)\lambda\Omega(\hat{\beta}) + t\lambda\Omega(\beta^*). \end{aligned}$$

Thus

$$\frac{R_n(\hat{\beta}) - R_n(\hat{\beta}_t)}{t} \le \lambda\Omega(\beta^*) - \lambda\Omega(\hat{\beta}).$$

By the G-convexity

$$R(\hat{\beta}_t) \le (1-t)R(\hat{\beta}) + tR(\beta^*) - tG\Big(\tau(\beta^* - \hat{\beta})\Big).$$

Therefore

$$R(\beta^*) - R(\hat{\beta}) \ge \frac{R(\hat{\beta}_t) - R(\hat{\beta})}{t} + G\Big(\tau(\beta^* - \hat{\beta})\Big).$$

- If

$$\frac{R(\hat{\beta}_t) - R(\hat{\beta})}{t} \ge -\lambda_\varepsilon\gamma_n - 2\lambda\Omega(\beta^*_{S^{*c}}) + \delta\underline{\lambda\Omega}(\hat{\beta} - \beta)$$

we see from the G-convexity

$$\begin{aligned} R(\beta) - R(\hat{\beta}) &\ge -\lambda_\varepsilon\gamma_n - 2\lambda\Omega(\beta^*_{S^{*c}}) + \delta\underline{\lambda\Omega}(\hat{\beta} - \beta^*) + G\Big(\tau(\beta^* - \hat{\beta})\Big) \\ &\ge -\lambda_\varepsilon\gamma_n - 2\lambda\Omega(\beta^*_{S^{*c}}) + \delta\underline{\lambda\Omega}(\hat{\beta} - \beta^*) \end{aligned}$$

and thus

$$\delta\underline{\lambda\Omega}(\hat{\beta} - \beta^*) + R(\hat{\beta}) \le R(\beta^*) + \lambda_\varepsilon\gamma_n + 2\lambda\Omega(\beta^*_{S^{*c}})$$

and we are done.

- From now on we assume

$$\frac{R(\hat{\beta}_t) - R(\hat{\beta})}{t} \le -\lambda_\varepsilon\gamma_n - 2\lambda\Omega(\beta^*_{S^{*c}}) + \delta\underline{\lambda\Omega}(\hat{\beta} - \beta^*).$$

Since

$$\frac{R_n(\hat{\beta}) - R_n(\hat{\beta}_t)}{t} \le \lambda\Omega(\beta^*) - \lambda\Omega(\hat{\beta})$$

we know that

$$\begin{aligned}
&\lambda_\varepsilon \gamma_n + 2\lambda\Omega(\beta^*_{S^{*c}}) \\
&\le \frac{R(\hat\beta) - R(\hat\beta_t)}{t} + \delta\underline{\lambda\Omega}(\hat\beta - \beta^*) \\
&= \frac{R_n(\hat\beta) - R_n(\hat\beta_t)}{t} - \frac{\left[R_n - R\right]\hat\beta - \left[R_n - R\right]\hat\beta_t}{t} + \delta\underline{\lambda\Omega}(\hat\beta - \beta^*) \\
&\le -\frac{\left[R_n - R\right]\hat\beta - \left[R_n - R\right]\hat\beta_t}{t} + \lambda\Omega(\beta^*) - \lambda\Omega(\hat\beta) + \delta\underline{\lambda\Omega}(\hat\beta - \beta^*) \\
&\le \lambda_\varepsilon \underline{\Omega}(\hat\beta - \beta^*) + \lambda_\varepsilon\gamma_n + \lambda\Omega(\beta^*) - \lambda\Omega(\hat\beta) + \delta\underline{\lambda\Omega}(\hat\beta - \beta^*) \\
&\le \lambda_\varepsilon(\Omega(\hat\beta_{S^*} - \beta^*_{S^*}) + \Omega^{S^{*c}}(\hat\beta_{S^{*c}} - \beta^*_{S^{*c}})) \\
&+ \lambda\Omega(\hat\beta_{S^*} - \beta^*_{S^*}) + \lambda\Omega(\beta^*_{S^{*c}}) - \lambda\Omega^{S^{*c}}(\hat\beta_{S^{*c}}) \\
&+ \lambda_\varepsilon\gamma_n + \delta\underline{\lambda\Omega}(\hat\beta - \beta^*) \\
&\le \lambda_\varepsilon(\Omega(\hat\beta_{S^*} - \beta^*_{S^*}) + \Omega^{S^{*c}}(\hat\beta_{S^{*c}} - \beta^*_{S^{*c}})) \\
&+ \lambda\Omega(\hat\beta_{S^*} - \beta^*_{S^*}) + \lambda\Omega(\beta^*_{S^{*c}}) - \lambda\Omega^{S^{*c}}(\hat\beta_{S^{*c}} - \beta^*_{S^{*c}}) + \lambda\Omega^{S^c}(\beta^*_{S^{*c}}) \\
&+ \lambda_\varepsilon\gamma_n + \delta\underline{\lambda\Omega}(\hat\beta - \beta^*) \\
&\le \lambda_\varepsilon(\Omega(\hat\beta_{S^*} - \beta^*_{S^*}) + \Omega^{S^{*c}}(\hat\beta_{S^{*c}} - \beta^*_{S^{*c}})) \\
&+ \lambda\Omega(\hat\beta_{S^*} - \beta^*_{S^*}) - \lambda\Omega^{S^{*c}}(\hat\beta_{S^{*c}} - \beta^*_{S^{*c}}) + 2\lambda\Omega(\beta^*_{S^{*c}}) \\
&+ \lambda_\varepsilon\gamma_n + \delta\underline{\lambda\Omega}(\hat\beta - \beta^*).
\end{aligned}$$

This gives

$$\Omega^{S^{*c}}(\hat\beta_{S^{*c}} - \beta_{S^{*c}}) \le L\Omega(\hat\beta_{S^*} - \beta^*_{S^*})$$

and hence

$$\Omega(\hat\beta_{S^*} - \beta^*_{S^*}) \le \Gamma(\tau, L, S^*)\tau(\hat\beta_{S^*} - \beta^*_{S^*}).$$

But then applying the dual conjugate inequality

$$
\begin{aligned}
&\lambda_\varepsilon \gamma_n + 2\lambda\Omega(\beta^*_{S^{*c}}) \\
&\le \frac{R(\hat\beta) - R(\hat\beta_t)}{t} + \delta\underline{\lambda\Omega}(\hat\beta - \beta^*) \\
&\le \overline{\lambda}\Gamma(\tau, L, S^*)\tau(\hat\beta_{S^*} - \beta^*_{S^*}) + \lambda_\varepsilon\gamma_n + 2\lambda\Omega(\beta^*_{S^{*c}}) \\
&\le G\Big(\tau(\beta^* - \hat\beta)\Big) + H\Big(\overline{\lambda}\Gamma(\tau, L, S^*)\Big) + \lambda_\varepsilon\gamma_n + 2\lambda\Omega(\beta^*_{S*c}) \\
&\le R(\beta^*) - R(\hat\beta) + \frac{R(\hat\beta) - R(\hat\beta_t)}{t} + H\Big(\overline{\lambda}\Gamma(L, \beta^*_{S^*}, \tau)\Big) \\
&+ \lambda_\varepsilon\gamma_n + 2\lambda\Omega(\beta^*_{S^{*c}}),
\end{aligned}
$$

where in the last step we used the G-convexity. In other words then

$$
\delta\underline{\lambda\Omega}(\hat\beta - \beta^*) + R(\hat\beta) \le R(\beta^*) + H\Big(\overline{\lambda}\Gamma(\tau, L, S^*)\Big) + \lambda_\varepsilon\gamma_n + 2\lambda\Omega(\beta^*_{S^{*c}}).
$$

Proof of Lemma 1

By a two-term Taylor expansion

$$
g((1-t)u + tv) = g(u) + t(v-u)\dot g(u) + \tfrac{1}{2}t^2(v-u)^2\ddot g(\tilde u),
$$

where $\tilde u$ is an intermediate point. Similarly

$$
g(v) = g(u) + (v-u)\dot g(u) + \tfrac{1}{2}(v-u)^2\ddot g(\bar u),
$$

where $\bar u$ is another intermediate point. Therefore

$$
(1-t)g(u) + tg(v) = g(u) + t(g(v) - g(u)) + (v-u)\dot g(u) + \tfrac{1}{2}t(v-u)^2\ddot g(\bar u).
$$

Taking the difference yields

$$
\begin{aligned}
g((1-t)u + tv) - (1-t)g(u) + tg(v) &= \tfrac{1}{2}t^2(v-u)^2\ddot g(\tilde u) - \tfrac{1}{2}t(v-u)^2\ddot g(\bar u) \\
&\le -\tfrac{1}{2}(1/\underline{C}^2 - \bar C^2 t)t(v-u)^2 \\
&= -\tfrac{1}{4}t(v-u)^2/\underline{C}^2.
\end{aligned}
$$

References

1. F. Bach, R. Jenatton, J. Mairal, G. Obozinski, Optimization with sparsity-inducing penalties. Found. Trends® Mach. Learn. **4**(1), 1–106 (2012)
2. F.R. Bach, Structured sparsity-inducing norms through submodular functions, in *Advances in Neural Information Processing Systems*, pp. 118–126 (2010)
3. D.P. Bertsekas, *Nonlinear Programming* (Athena Scientific, Belmont, 1999)
4. P.J. Bickel, Y. Ritov, A.B. Tsybakov, Simultaneous analysis of lasso and dantzig selector. Ann. Statist. **37**(4), 1705–1732 (2009)
5. M. Bogdan, E. van den Berg, W. Su, E. Candès, Statistical estimation and testing via the sorted l1 norm. arXiv preprint arXiv:1310.1969 (2013)
6. P. Bühlmann, S. van de Geer, *Statistics for High-Dimensional Data: Methods, Theory and Applications* (Springer Science & Business Media, 2011)
7. F. Bunea, A. Tsybakov, M. Wegkamp, Sparsity oracle inequalities for the lasso. Electron. J. Stat. **1**, 169–194 (2007)
8. E. Candès, T. Tao, The Dantzig selector: Statistical estimation when p is much larger than n. Ann. Statist. **35**(6), 2313–2351 (2007)
9. A. Elsener, S. van de Geer, Robust low-rank matrix estimation. Ann. Statist. **46**(6B), 3481–3509 (2018)
10. A. Elsener, S. van de Geer, Sharp oracle inequalities for stationary points of nonconvex penalized M-estimators. IEEE Trans. Inform. Theory **65**(3), 1452–1472 (2019)
11. I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning* (MIT Press, 2016). http://www.deeplearningbook.org
12. J. Janková, S. van de Geer, De-biased sparse PCA: Inference and testing for eigenstructure of large covariance matrices. arXiv preprint arXiv:1801.10567 (2018)
13. R. Jenatton, J.-Y. Audibert, F. Bach, Structured variable selection with sparsity-inducing norms. J. Mach. Learn. Res. **12**, 2777–2824 (2011)
14. I.M. Johnstone, A.Yu. Lu, On consistency and sparsity for principal components analysis in high dimensions. J. Amer. Statist. Assoc. **104**(486), 682–693 (2009)
15. V. Koltchinskii, Oracle Inequalities in Empirical Risk Minimization and Sparse Recovery Problems: École dÉté de Probabilités de Saint-Flour XXXVIII-2008. Lecture Notes in Mathematics. Springer (2011)
16. V. Koltchinskii, K. Lounici, A.B. Tsybakov, Nuclear-norm penalization and optimal rates for noisy low-rank matrix completion. Ann. Statist. **39**(5), 2302–2329 (2011)
17. M. Ledoux, M. Talagrand, *Probability in Banach Spaces: Isoperimetry and Processes* vol. 23. (Springer Science & Business Media, 1991)
18. P.-L. Loh, Statistical consistency and asymptotic normality for high-dimensional robust M-estimators. Ann. Statist. **45**(2), 866–896 (2017)
19. P.-L. Loh, M.J. Wainwright, High-dimensional regression with noisy and missing data: Provable guarantees with nonconvexity. Ann. Statist. **40**(3), 1637–1664 (2012)
20. P.-L. Loh, M.J. Wainwright, Regularized M-estimators with Nonconvexity. J. Mach. Learn. Res. **16**, 559–616 (2015)
21. P.-L. Loh, M.J. Wainwright, Support recovery without incoherence: a case for nonconvex regularization. Ann. Statist. **45**(6), 2455–2482 (2017)
22. A. Maurer, M. Pontil, Structured sparsity and generalization. J. Mach. Learn. Res. **13**, 671–690 (2012)
23. Yu.S. Mei, Bai, A. Montanari, The landscape of empirical risk for nonconvex losses. Ann. Statist. **46**(6A), 2747–2774 (2018)
24. J. Morales, C.A. Micchelli, M. Pontil, A family of penalty functions for structured sparsity. Adv. Neural Inf. Process. Syst. **23**, 1612–1623 (2010)
25. P. Rigollet, A. Tsybakov, Exponential screening and optimal rates of sparse estimation. Ann. Statist. **39**(2), 731–771 (2011)
26. P. Rigollet, High-dimensional statistics

27. B. Stucky, S. van de Geer, Sharp oracle inequalities for square root regularization. J. Mach. Learn. Res. **18**(67), 1–29 (2017)
28. R. Tibshirani, Regression shrinkage and selection via the lasso. J. R. Stat. Soc. Ser. B. Stat. Methodol. **58**, 267–288 (1996)
29. S. van de Geer, Weakly decomposable regularization penalties and structured sparsity. Scand. J. Stat. **41**(1), 72–86 (2014)
30. S. van de Geer, *Estimation and Testing under Sparsity: École dÉté de Probabilités de Saint-Flour XLV-2015* (Lecture Notes in Mathematics. Springer, 2016)

Median-Truncated Gradient Descent: A Robust and Scalable Nonconvex Approach for Signal Estimation

Yuejie Chi, Yuanxin Li, Huishuai Zhang and Yingbin Liang

Abstract Recent work has demonstrated the effectiveness of gradient descent for directly estimating high-dimensional signals via nonconvex optimization in a globally convergent manner using a proper initialization. However, the performance is highly sensitive in the presence of adversarial outliers that may take arbitrary values. In this chapter, we introduce the median-Truncated Gradient Descent (median-TGD) algorithm to improve the robustness of gradient descent against outliers, and apply it to two celebrated problems: low-rank matrix recovery and phase retrieval. Median-TGD truncates the contributions of samples that deviate significantly from the *sample median* in each iteration in order to stabilize the search direction. Encouragingly, when initialized in a neighborhood of the ground truth known as the basin of attraction, median-TGD converges to the ground truth at a linear rate under Gaussian designs with a near-optimal number of measurements, even when a constant fraction of the measurements are arbitrarily corrupted. In addition, we introduce a new median-truncated spectral method that ensures an initialization in the basin of attraction. The stability against additional dense bounded noise is also established. Numerical experiments are provided to validate the superior performance of median-TGD.

Y. Chi (✉) · Y. Li
Department of Electrical and Computer Engineering, Carnegie Mellon University,
5000 Forbes Avenue, Pittsburgh, PA 15213, USA
e-mail: yuejie.chi@cmu.edu

Y. Li
e-mail: yuanxinl@andrew.cmu.edu

H. Zhang
Microsoft Research Asia, 5 Danling Street Microsoft Tower 2, Haidian District,
Beijing 100080, China
e-mail: Huishuai.Zhang@microsoft.com

Y. Liang
Department of Electrical and Computer Engineering, The Ohio State University,
2015 Neil Avenue, Columbus, OH 43210, USA
e-mail: liang.889@osu.edu

H. Boche et al. (eds.), *Compressed Sensing and Its Applications*,
Applied and Numerical Harmonic Analysis,
https://doi.org/10.1007/978-3-319-73074-5_8

1 Introduction

For many problems in science and engineering, one collects measurements $\{y_i\}_{i=1}^m$ of some unknown object $\boldsymbol{x}$, and aims to recover the object via solving an empirical risk minimization problem:

$$\hat{\boldsymbol{x}} = \mathrm{argmin}_{\mathbf{z}} \frac{1}{2m} \sum_{i=1}^{m} \ell(\mathbf{z}; y_i), \tag{1}$$

where $\ell(\mathbf{z}; y_i)$ is the sample loss function, for example, the negative log-likelihood function. This problem often is nonconvex, making it challenging to solve in a globally optimal manner in general. Indeed, it is known that even for optimizing a single-neuron model with the squared loss and the logistic activation function, there may exist exponentially many local minima [1]. Recently, there has been a surge of activities for studying the performance of simple iterative methods such as gradient descent for statistical estimation with nonconvex objective functions. Encouragingly, with the help of certain statistical models of data, strong global convergence guarantees may be possible by analyzing the average-case performance for certain benign nonconvex problems, including but not limited to, low-rank matrix recovery/completion [2], phase retrieval [3], dictionary learning [4], and blind deconvolution [5], to name a few. A typical result states that gradient descent with a proper initialization is guaranteed to converge to the ground truth with high probability, as soon as the sample size is large enough, under certain statistical models of data generation.

In practice, it is quite typical that measurements may suffer from outliers that need to be addressed carefully. There are many situations where outliers arise, such as detector failures, recording errors, and missing data, possibly in an adversarial fashion with outliers taking arbitrary values. Unfortunately, the vanilla gradient descent algorithm, though adopted widely, is very sensitive to the presence of even a single outlier, as the outliers can perturb the search directions arbitrarily. Therefore, it is greatly desirable to develop fast and robust alternatives that are globally convergent in a provable manner even with a large number of adversarial outliers.

This chapter introduces a median truncation strategy to robustify the vanilla gradient descent approach, which includes careful modifications on both the initialization and the local search procedures. As it is widely known, the sample median is a more robust quantity vis-à-vis outliers, compared with the sample mean. It requires half of the samples to be outliers in order to perturb the sample median arbitrarily, while only one sample suffices to perturb the sample mean [6]. Therefore, the median becomes an ideal quantity to illuminate which samples are likely to be outliers and therefore should be eliminated during the gradient descent updates. The new approach, called median-Truncated Gradient Descent (median-TGD), starts with formulating an initialization using a truncated spectral method, where only samples whose absolute values are not too deviated from the sample median, are included. Next, it follows by a truncated gradient update, where only samples whose measurement residuals evaluated at the current estimate are not too deviated from the sample median are included.

This leads to an adaptive and iteration-varying strategy to mitigate the effects of outliers. The effectiveness of median-TGD is illustrated on two important problems, low-rank matrix recovery [7, 8] and phase retrieval [9], where median-TGD provably tolerates a constant fraction of outliers at a near-optimal sample complexity up to some logarithmic factors. Computationally, because the sample median can be computed in a linear time [10], median-TGD shares a similar attractive computational cost as the vanilla gradient descent while being a lot more robust.

The remainder of this chapter is organized as follows. Section 2 describes the general recipe of median-TGD. Section 3 adopts median-TGD to low-rank matrix recovery and describes its performance guarantee. Section 4 adopts median-TGD to phase retrieval and describes its performance guarantee. Section 5 discusses the main ingredients for analysis and highlights a few properties of median. Section 6 provides numerical evidence on the superior performance of median-TGD in the presence of outliers. Section 7 reviews the related literature. Finally, we conclude the chapter in Sect. 8.

Notations: We denote vectors by boldface lowercase letters and matrices by boldface uppercase letters. The notations $\mathbf{A}^T$, $\|\mathbf{A}\|$, and $\|\mathbf{A}\|_F$ represent the transpose, the spectral norm, and the Frobenius norm of a matrix $\mathbf{A}$, respectively. We denote the kth singular value of $\mathbf{A}$ as $\sigma_k(\mathbf{A})$, and the kth eigenvalue as $\lambda_k(\mathbf{A})$. For a vector $\mathbf{y} \in \mathbb{R}^n$, $\mathrm{med}(\mathbf{y})$ denotes the median of the entries in $\mathbf{y}$, and $|\mathbf{y}|$ denotes the vector that contains its entry-wise absolute values. The (k, t)th entry of a matrix $\mathbf{A}$ is denoted as $\mathbf{A}_{k,t}$. Besides, the inner product between two matrices $\mathbf{A}$ and $\mathbf{B}$ is defined as $\langle \mathbf{A}, \mathbf{B} \rangle = \mathrm{Tr}\left(\mathbf{B}^T\mathbf{A}\right)$, where $\mathrm{Tr}(\cdot)$ denotes the trace of a matrix. The indicator function of an event A is denoted as $\mathbb{I}_A$, which equals 1 if A is true and 0 otherwise. In addition, we use $C, c_1, c_2, \ldots$ with different subscripts to represent universal constants, whose values may change from line to line. The notation $f(n) = \Omega(g(n))$ or $f(n) \gtrsim g(n)$ means that there exists a constant $c > 0$ such that $|f(n)| \geq c|g(n)|$ for all sufficiently large n.

2 Median-Truncated Gradient Descent

In this section, we describe a unified framework to outline the approach of median-TGD. Consider that one collects measurements y_i's that are modeled as

$$y_i \approx f_i(\mathbf{x}), \qquad 1 \leq i \leq m,$$

where $f_i(\mathbf{x})$'s are independent and identically distributed (i.i.d.) given $\mathbf{x}$, and the randomness can be due to the sampling of measurement operators. Denote the index set of corrupted measurements by $\mathcal{S}$, and correspondingly, the index set of clean measurements is given as the complementary set $\mathcal{S}^c$. Mathematically, the measurements $\mathbf{y} = \{y_i\}_{i=1}^m$ are given as

$$y_i = \begin{cases} f_i(\mathbf{x}) + w_i, & \text{if } \ i \in \mathcal{S}^c; \\ \eta_i + w_i, & \text{if } \ i \in \mathcal{S}, \end{cases} \tag{2}$$

where $\boldsymbol{\eta} = \{\eta_i\}_{i\in\mathcal{S}}$ is the set of outliers that can take arbitrary values. Denote the cardinality of $\mathcal{S}$ as $|\mathcal{S}| = s \cdot m$, where $0 \le s < 1$ is the fraction of outliers. Furthermore, $\mathbf{w} = \{w_i\}_{i=1}^m$ is the additional deterministic dense bounded noise that satisfies $\|\mathbf{w}\|_\infty \le c_w$ for some small c_w.[1]

A natural strategy for estimating $\mathbf{x}$ is to solve (1) with a carefully selected loss function. For simplicity, we limit our discussions to the quadratic loss function, i.e., $\ell(\mathbf{z}; y_i) = [y_i - f_i(\mathbf{x})]^2$, but the approach extends to other loss functions as well. Due to the presence of outliers, the signal of interest may no longer be the global optima of (1). Therefore, an ideal approach is to minimize an *oracle* loss function,

$$f_{\text{oracle}}(\mathbf{z}) = \frac{1}{2m} \sum_{i \in \mathcal{S}^c} \ell(\mathbf{z}; y_i), \tag{3}$$

which aims to minimize the quadratic loss over only the *clean* measurements. Nevertheless, it is impossible to minimize $f_{\text{oracle}}(\mathbf{z})$ directly, since the oracle information regarding the support of outliers is absent. Moreover, the loss function can be nonconvex for many interesting choices of $f_i(\mathbf{x})$, adding difficulty to its global optimization.

Our key strategy is to prune the bad samples adaptively and iteratively, using a gradient descent procedure that proceeds as follows:

$$\mathbf{z}^{(t+1)} = \mathbf{z}^{(t)} - \frac{\mu}{m} \sum_{i \in T_{t+1}} \nabla \ell(\mathbf{z}^{(t)}; y_i). \tag{4}$$

where $\mathbf{z}^{(t)}$ denotes the tth iterate of gradient descent, $\nabla \ell(\mathbf{z}^{(t)}; y_i)$ is the gradient of $\ell(\mathbf{z}^{(t)}; y_i)$, and μ is the step size, for $t = 0, 1, \ldots$. In each iteration, only a subset T_{t+1} of data-dependent and iteration-varying samples contributes to the search direction. But how to select the set T_{t+1}?

Note that the gradient of the loss function typically contains the term $(y_i - f_i(\mathbf{z}^{(t)}))$, which measures the residual using the current iterate. With y_i being corrupted by arbitrarily large outliers, the gradient can deviate the search direction from the signal arbitrarily. Inspired by the utility of *median* to combat outliers in robust statistics [6], we prune samples whose gradient components $\nabla \ell(\mathbf{z}^{(t)}; y_i)$ are much larger than the *median* to control the search direction of each update. This yields the main ingredient of the *median-truncated gradient descent* (median-TGD) update rule, i.e., for each iterate $t \ge 0$:

$$T_{t+1} := \left\{ i : |y_i - f_i(\mathbf{z}^{(t)})| \lesssim \mathsf{med}(\{|y_i - f_i(\mathbf{z}^{(t)})|\}_{i=1}^m) \right\},$$

[1] It is straightforward to handle stochastic noise such as Gaussian noise, by noticing that its infinity norm is bounded with high probability.

where $\mathsf{med}(\cdot)$ denotes the sample median. The robust property of median lies in the fact that the median cannot be arbitrarily perturbed unless the outliers dominate the inliers [6]. This is in sharp contrast to the sample mean, which can be made arbitrarily large even by a single outlier. Thus, using the sample median in the truncation rule can effectively remove the impact of outliers. Finally, there still left the question of initialization, which is critical to the success of the algorithm. We use the spectral method, i.e., initialize $\mathbf{z}^{(0)}$ with the help of a certain surrogate data matrix,

$$\mathbf{Y} = \frac{1}{m} \sum_{i \in T_0} y_i \mathbf{B}_i, \tag{5}$$

for some choice of $\mathbf{B}_i$ that depends on the form of $f_i(\mathbf{x})$. Again, the set T_0 includes only a subset of samples whose values are not excessively large compared with the sample median of the measurements in terms of absolute values, given as

$$T_0 = \left\{ i : |y_i| \lesssim \mathsf{med}(\{|y_i|\}_{i=1}^m) \right\}.$$

Putting things together (the update rule (4) and the initialization (5)), we obtain the new median-TGD algorithm, based on applying the median truncation strategy. The median-TGD algorithm does not assume a priori knowledge of the outliers, such as their existence or the number of outliers, and therefore can be used in an oblivious fashion. In the next two sections, we apply median-TGD to two important problems, low-rank matrix recovery and phase retrieval, respectively, and demonstrate the appealing theoretical properties.

3 Robust Low-Rank Matrix Recovery

In this section, we apply median-TGD to the problem of robust low-rank matrix recovery. Low-rank matrix recovery is a problem of great interest in applications such as collaborative filtering, signal processing, and computer vision. A considerable amount of work has been done on low-rank matrix recovery in recent years, where it is shown that low-rank matrices can be recovered accurately and efficiently from much fewer observations than their ambient dimensions [11–16]. An extensive overview of low-rank matrix recovery can be found in [17].

Let $\mathbf{M} \in \mathbb{R}^{n \times n}$ be a rank r positive semidefinite matrix that can be written as

$$\mathbf{M} = \mathbf{X}\mathbf{X}^T, \tag{6}$$

where $\mathbf{X} \in \mathbb{R}^{n \times r}$ is the low-rank factor of $\mathbf{M}$.[2] Define the condition number and the *average* condition number of $\mathbf{M}$ as $\kappa = \lambda_1(\mathbf{M})/\lambda_r(\mathbf{M})$, and $\bar{\kappa} = \|\mathbf{M}\|_F/(\sqrt{r}\lambda_r(\mathbf{M}))$. Clearly, $\bar{\kappa} \leq \kappa$. Also, as a useful fact, we have $\lambda_i(\mathbf{M}) = \sigma_i^2(\mathbf{X})$, $i = 1, \ldots, r$.

[2] Our discussions can be extended to the rectangular case, see [8].

Let m be the number of measurements, and the set of sensing matrices is given as $\{\mathbf{A}_i\}_{i=1}^m$, where $\mathbf{A}_i \in \mathbb{R}^{n\times n}$ is the ith symmetric sensing matrix. In particular, each $\mathbf{A}_i$ is generated i.i.d. from Gaussian orthogonal ensemble (GOE), with $(\mathbf{A}_i)_{k,k} \sim \mathcal{N}(0,2)$, $(\mathbf{A}_i)_{k,t} \sim \mathcal{N}(0,1)$ for $k < t$, and $(\mathbf{A}_i)_{k,t} = (\mathbf{A}_i)_{t,k}$. Specialize the measurement model (2) to low-rank matrix sensing, we have

$$y_i = \begin{cases} \langle \mathbf{A}_i, \mathbf{M}\rangle + w_i, & \text{if} \quad i \in \mathcal{S}^c \\ \eta_i + w_i, & \text{if} \quad i \in \mathcal{S} \end{cases}. \tag{7}$$

To simplify notations, we define the linear maps $\mathcal{A}_i(\mathbf{W}) = \{\mathbb{R}^{n\times n} \mapsto \mathbb{R} : \langle \mathbf{A}_i, \mathbf{W}\rangle\}$, and $\mathcal{A}(\mathbf{W}) = \{\mathbb{R}^{n\times n} \mapsto \mathbb{R}^m : \{\mathcal{A}_i(\mathbf{W})\}_{i=1}^m\}$.

Instead of recovering $\mathbf{M}$, we aim to directly recover its low-rank factor $\mathbf{X}$ from the corrupted measurements $\mathbf{y}$, without a priori knowledge of the outliers, in a computationally efficient and provably accurate manner. It is straightforward to see that for any orthonormal matrix $\mathbf{P} \in \mathbb{R}^{r\times r}$, we have $(\mathbf{XP})(\mathbf{XP})^T = \mathbf{XX}^T$, and consequently, $\mathbf{X}$ can be recovered only up to orthonormal transformations. Hence, we measure the estimation accuracy by taking this into consideration. Given $\mathbf{U} \in \mathbb{R}^{n\times r}$, the distance between $\mathbf{U}$ and $\mathbf{X}$ is measured as

$$\text{dist}\,(\mathbf{U}, \mathbf{X}) = \min_{\mathbf{P}\in\mathbb{R}^{r\times r}, \mathbf{PP}^T=\mathbf{I}} \|\mathbf{U} - \mathbf{XP}\|_F .$$

3.1 Median-TGD for Robust Low-Rank Matrix Recovery

We instantiate the approach of median-TGD to low-rank matrix recovery, by setting the sample loss function as

$$\ell(\mathbf{U}; y_i) := \left|y_i - \mathcal{A}_i(\mathbf{UU}^T)\right|^2,$$

where $\mathbf{U} \in \mathbb{R}^{n\times r}$. In each iteration, only a subset of the samples contribute to the search direction:

$$\mathbf{U}^{(t+1)} := \mathbf{U}^{(t)} - \frac{\mu_t}{\|\mathbf{U}^{(0)}\|^2} \nabla f_{tr}(\mathbf{U}^{(t)}), \tag{8}$$

where we denote $\nabla f_{tr}(\mathbf{U}_t) = \frac{1}{m}\sum_{i\in\mathcal{E}^t} \nabla \ell(\mathbf{U}^{(t)}; y_i)$ as the truncated gradient. In (8), μ_t denotes the step size, $\mathbf{U}^{(0)}$ is the initialization, and $\nabla \ell(\mathbf{U}; y_i)$ is the gradient of $\ell(\mathbf{U}; y_i)$ given as

$$\nabla \ell(\mathbf{U}; y_i) = (\mathcal{A}_i(\mathbf{UU}^T) - y_i)\mathbf{A}_i\mathbf{U}.$$

Importantly, the set $\mathcal{E}^t$ is set adaptively to rule out outliers. Denote the residual of the ith measurement at the tth iteration as

$$r_i^{(t)} = y_i - \mathcal{A}_i(\mathbf{U}^{(t)}\mathbf{U}^{(t)T}), \quad i = 1, 2, \ldots, m,$$

and $\boldsymbol{r}^{(t)} = [r_1^{(t)}, r_2^{(t)}, \ldots, r_m^{(t)}]^T = \mathbf{y} - \mathcal{A}(\mathbf{U}^{(t)}\mathbf{U}^{(t)T})$. Then the set $\mathcal{E}^t$ is defined as

$$\mathcal{E}^t = \left\{ i \middle| |r_i^{(t)}| \leq \alpha_h \cdot \mathrm{med}\{|\boldsymbol{r}^{(t)}|\} \right\},$$

where α_h is some constant. In other words, only samples whose current absolute residuals are not too deviated from the sample median of the absolute residuals are included in the gradient update. As the estimate $\mathbf{U}^{(t)}$ gets more accurate, we expect that the set $\mathcal{E}^t$ gets closer to the oracle set $\mathcal{S}^c$, and hence the gradient search is more accurate. Note that the set $\mathcal{E}^t$ varies per iteration, and therefore can adaptively prune the outliers.

For initialization, we adopt a truncated spectral method, which uses the top eigenvectors of a sample-weighted surrogate matrix, where again only the samples whose absolute values do not significantly digress from the sample median are included. To avoid statistical dependence in the theoretical analysis, we split the samples by using the sample median of m_2 samples to estimate $\|\mathbf{M}\|_F$, and then using the rest of the samples to construct the truncated surrogate matrix to perform a spectral initialization. In practice, we find that this sample split is unnecessary, as demonstrated in the numerical simulations.

The details of median-TGD are provided in Algorithm 1, where the stopping criterion is simply set as reaching a preset maximum number of iterations. In practice, it is also possible to set the stopping criteria by examining the progress between iterations.

3.2 Theoretical Guarantees

Encouragingly, when initialized in a basin of attraction close to the ground truth, median-TGD converges globally at a linear rate under the GOE model with an order of $\mathcal{O}(nr \log n)$ measurements, even when a constant fraction of the measurements is arbitrarily corrupted, which is near-optimal up to a logarithmic factor. In addition, the truncated spectral method ensures an initialization in the basin of attraction with an order of $\mathcal{O}(nr^2 \log n \log^2 r)$ measurements when a fraction of $1/\sqrt{r}$ measurements are arbitrarily corrupted. In the case when the rank is a small constant, median-TGD provably tolerates a constant fraction of outliers with $\mathcal{O}(n \log n)$ measurements, which is much smaller than the ambient dimension n^2 of the matrix. Furthermore, the stability of median-TGD against additional dense bounded noise is also established.

Theorem 1 summarizes the performance guarantee of median-TGD in Algorithm 1 for low-rank matrix recovery in the presence of both sparse arbitrary outliers and dense bounded noise when initialized around a proper neighborhood around the ground truth. The proof can be found in [8].

Algorithm 1: median-TGD for robust low-rank matrix recovery

```
Parameters: Thresholds $\alpha_y$ and $\alpha_h$, step size $\mu_t$, average condition number bound $\bar{\kappa}_0$, and rank $r$.
Input: The measurements $\mathbf{y} = \{y_i\}_{i=1}^{m}$, and the sensing matrices $\{\mathbf{A}_i\}_{i=1}^{m}$.
Initialization:
(1) Set $\mathbf{y}_1 = \{y_i\}_{i=1}^{m_1}$ and $\mathbf{y}_2 = \{y_i\}_{i=m_1+1}^{m}$, where $m_1 = \lceil m/2 \rceil$ and $m_2 = m - m_1$.
(2) Initialize $\mathbf{U}^{(0)} = \mathbf{Z}\boldsymbol{\Sigma}$, where the columns of $\mathbf{Z}$ contain the normalized eigenvectors
    corresponding to the $r$ largest eigenvalues in terms of absolute values, i.e.
    $|\lambda_1(\mathbf{Y})| \geq |\lambda_2(\mathbf{Y})| \geq \cdots \geq |\lambda_r(\mathbf{Y})|$, of the matrix
        $\mathbf{Y} = \frac{1}{m_1} \sum_{i=1}^{m_1} y_i \mathbf{A}_i \mathbb{I}_{\{|y_i| \leq \alpha_y \cdot \mathrm{med}(|\mathbf{y}_2|)\}},$    (9)
    and $\boldsymbol{\Sigma}$ is an $r \times r$ diagonal matrix, with $\boldsymbol{\Sigma}_{i,i} = \sqrt{|\lambda_i(\mathbf{Y})|/2}$, $i = 1, 2, \ldots, r$.
Gradient Loop: For $t = 0 : 1 : T-1$ do
        $\mathbf{U}^{(t+1)} = \mathbf{U}^{(t)} - \frac{\mu_t}{\left\|\mathbf{U}^{(0)}\right\|^2} \cdot \frac{1}{m} \sum_{i=1}^{m} \left( \mathcal{A}_i(\mathbf{U}^{(t)}\mathbf{U}^{(t)T}) - y_i \right) \mathbf{A}_i \mathbf{U}^{(t)} \mathbb{I}_{\mathcal{E}_i^t},$    (10)
    where
        $\mathcal{E}_i^t = \left\{ \left| y_i - \mathcal{A}_i(\mathbf{U}^{(t)}\mathbf{U}^{(t)T}) \right| \leq \alpha_h \cdot \mathrm{med}\left( \left| \mathbf{y} - \mathcal{A}\left( \mathbf{U}^{(t)}\mathbf{U}^{(t)T} \right) \right| \right) \right\}.$    (11)
Output: $\hat{\mathbf{X}} = \mathbf{U}^{(T)}$.
```

Theorem 1 *Consider the measurement model* (7) *with* $\|\mathbf{w}\|_\infty \leq c_w \lambda_r(\mathbf{M})$ *for a sufficiently small constant* c_w*. Suppose that the initialization* $\mathbf{U}^{(0)}$ *satisfies*

$$\mathrm{dist}\left(\mathbf{U}^{(0)}, \mathbf{X}\right) \leq \frac{1}{12} \sigma_r(\mathbf{X}).$$

Set $\alpha_h = 6$*. There exist some constants* $0 < s_0 < 1$*,* $c_0 > 1, c_1 > 1$ *such that with probability at least* $1 - e^{-c_1 m}$*, if* $s \leq s_0$*, and* $m \geq c_1 nr \log n$*, then there exists a constant* $\mu \leq \frac{1}{600}$*, such that with* $\mu_t \leq \mu$*, the estimates of median-TGD satisfy*

$$\mathrm{dist}\left(\mathbf{U}^{(t)}, \mathbf{X}\right) \lesssim \frac{\|\mathbf{w}\|_\infty}{\sigma_r(\mathbf{X})} + \left(1 - \frac{2\mu}{5\kappa}\right)^{t/2} \mathrm{dist}\left(\mathbf{U}^{(0)}, \mathbf{X}\right).$$

Theorem 1 suggests that if the initialization $\mathbf{U}^{(0)}$ lies in the basin of attraction, then median-TGD converges to the ground truth at a linear rate as long as the number m of measurements is in the order of $\mathcal{O}(nr \log n)$, even when a constant fraction of measurements are corrupted arbitrarily. In comparisons, the vanilla gradient descent algorithm by Tu et al. [18] achieves the same convergence rate in a similar basin of attraction, with an order of $\mathcal{O}(nr)$ measurements using outlier-free measurements. Therefore, median-TGD achieves robustness up to a constant fraction of outliers with a slight price of an additional logarithmic factor in the sample complexity.

Furthermore, Theorem 1 justifies the stability of median-TGD when the noise level $\|\boldsymbol{w}\|_\infty$ is not too large.

Theorem 2 provides that the truncated spectral method provides an initialization in the basin of attraction with high probability.

Theorem 2 *Assume the measurement model* (7) *with* $\|\mathbf{w}\|_\infty \leq c_m \lambda_r(\mathbf{M})$ *for a sufficiently small constant* c_m*, and* $\bar{\kappa} \leq \bar{\kappa}_0$*. Set* $\alpha_y = 2\log(r^{1/4}\bar{\kappa}_0^{1/2} + 20)$*. There exist some constants* $0 < s_1 < 1$ *and* $c_2, c_3, c_4 > 1$ *such that with probability at least* $1 - n^{-c_2} - \exp(-c_3 m)$*, if* $s \leq \frac{s_1}{\sqrt{r}\bar{\kappa}}$*, and* $m \geq c_4 \alpha_y^2 \bar{\kappa}^2 n r^2 \log n$*, then we have*

$$\operatorname{dist}\left(\mathbf{U}^{(0)}, \mathbf{X}\right) \leq \frac{1}{12}\sigma_r(\mathbf{X}).$$

Theorem 2 suggests that the proposed initialization scheme is guaranteed to obtain a valid initialization in the basin of attraction with an order of $\mathcal{O}(nr^2 \log n \log^2 r)$ measurements when a fraction of $1/\sqrt{r}$ measurements are arbitrarily corrupted, assuming the condition number κ is a small constant. In comparisons, in the outlier-free setting, Tu et al. [18] requires an order of $\mathcal{O}(nr^2\kappa^2)$ measurements for a one-step spectral initialization, which is closest to our scheme. Therefore, our initialization achieves robustness to a $1/\sqrt{r}$ fraction of outliers at a slight price of additional logarithmic factors in the sample complexity. Finally, we note that the parameter bounds in all theorems, including α_h, α_y, and μ, are not optimized for performance, but mainly selected to establish the theoretical guarantees.

4 Robust Phase Retrieval

In this section, we apply median-TGD to the problem of robust phase retrieval. Phase retrieval is a classical problem in signal processing, optics, and machine learning that has a wide range of applications such as X-ray crystallography [19], ptychography, and astronomical imaging [20]. Mathematically, it is formulated as recovering a signal $\mathbf{x} \in \mathbb{R}^n$ from the magnitudes of its linear measurements.[3] Consider the following model for phase retrieval, where the measurements are corrupted by not only sparse arbitrary outliers but also dense bounded noise. Under such a model, the measurements are given as

$$y_i = \left|\mathbf{a}_i^T \mathbf{x}\right| + w_i + \eta_i, \quad i = 1, \ldots, m, \tag{12}$$

where $\mathbf{x} \in \mathbb{R}^n$ is the unknown signal, $\mathbf{a}_i \in \mathbb{R}^n$ is the ith measurement vector composed of $i.i.d.$ Gaussian entries distributed as $\mathcal{N}(0, 1)$, and $\eta_i \in \mathbb{R}$ for $i = 1, \ldots, m$ are outliers with arbitrary values satisfying $\|\boldsymbol{\eta}\|_0 \leq s \cdot m$, where s is the fraction of outliers, and $\mathbf{w} = \{w_i\}_{i=1}^m$ is the bounded noise satisfying $\|\mathbf{w}\|_\infty \leq c\|\mathbf{x}\|$ for some universal constant c.

[3] The algorithm can be used to estimate complex-valued signals as well.

It is straightforward to observe that changing the sign of the signal does not affect the measurements. The goal is to recover the signal $\mathbf{x}$, up to a global sign difference, from the measurements $\mathbf{y} = \{y_i\}_{i=1}^m$ and the measurement vectors $\{\mathbf{a}_i\}_{i=1}^m$. To this end, we define the Euclidean distance between two vectors up to a global sign difference as the performance metric,

$$\mathrm{dist}(\mathbf{z}, \mathbf{x}) := \min\{\|\mathbf{z} + \mathbf{x}\|, \|\mathbf{z} - \mathbf{x}\|\}. \tag{13}$$

4.1 Median-TGD for Robust Phase Retrieval

We instantiate the approach of median-TGD to phase retrieval, by setting the sample loss function as the quadratic loss of the amplitudes:

$$\ell(\mathbf{z}; y_i) = \left(y_i - |\mathbf{a}_i^T \mathbf{z}|\right)^2. \tag{14}$$

Though (14) is not smooth everywhere, it has been argued in [21] that the loss function (14) resembles more closely to the quadratic loss when the phase information is available, and has a more amenable curvature for the convergence of the gradient descent algorithms.

In each iteration, only a subset of the samples contributes to the search direction:

$$\mathbf{z}^{(t+1)} := \mathbf{z}^{(t)} - \frac{\mu_t}{\|\mathbf{z}^{(0)}\|} \nabla f_{tr}(\mathbf{z}^{(t)}), \tag{15}$$

where $\nabla_{tr} f(\mathbf{U}_t) = \sum_{i \in \mathcal{E}^t} \ell(\mathbf{z}^{(t)}; y_i)$ denotes the truncated gradient. In (8), μ_t denotes the step size, and $\mathbf{z}^{(0)}$ denotes the initialization. Moreover, $\nabla \ell(\mathbf{z}; y_i)$ is the gradient of $\ell(\mathbf{z}; y_i)$ given as

$$\nabla \ell(\mathbf{z}; y_i) = \left(\mathbf{a}_i^T \mathbf{z}^{(t)} - y_i \cdot \frac{\mathbf{a}_i^T \mathbf{z}^{(t)}}{|\mathbf{a}_i^T \mathbf{z}^{(t)}|}\right) \mathbf{a}_i.$$

Importantly, the set $\mathcal{E}^t$ is set adaptively to rule out outliers. Denote the residual of the ith measurement at the tth iteration as

$$r_i^{(t)} = y_i - |\mathbf{a}_i^T \mathbf{z}^{(t)}|, \quad i = 1, 2, \ldots, m,$$

and $\boldsymbol{r}^{(t)} = [r_1^{(t)}, r_2^{(t)}, \ldots, r_m^{(t)}]^T = \mathbf{y} - |\mathbf{A}\mathbf{z}^{(t)}|$. Then the set $\mathcal{E}^t$ is defined as

$$\mathcal{E}^t = \left\{ i \middle| |r_i^{(t)}| \leq \alpha_h \cdot \mathrm{med}\{|\boldsymbol{r}^{(t)}|\} \right\},$$

where α_h is some constant. Similarly, as the estimate $\mathbf{z}^{(t)}$ gets more accurate, we expect that the set $\mathcal{E}^t$ gets closer to the oracle set $\mathcal{S}^c$, and hence the gradient search is more accurate.

For initialization, we adopt a truncated spectral method, which uses the rescaled top eigenvectors of a sample-weighted surrogate matrix, where again only the samples whose absolute values do not significantly digress from the sample median are included. The details of the median-TGD algorithm are described in Algorithm 2. The tuning parameters can be set as $\mu_t := \mu = 0.8$ and the truncation threshold as $\alpha_h = 5$.

Algorithm 2: median-TGD for robust phase retrieval

Input: $\mathbf{y} = \{y_i\}_{i=1}^m$, $\{\mathbf{a}_i\}_{i=1}^m$;

Parameters: threshold α_h, and step size μ;

Initialization: Let $\mathbf{z}^{(0)} = \lambda_0 \tilde{\mathbf{z}}$, where $\lambda_0 = \mathsf{med}(\mathbf{y})/0.455$ and $\tilde{\mathbf{z}}$ is the leading eigenvector of

$$\mathbf{Y} := \frac{1}{m}\sum_{i=1}^{m} y_i \boldsymbol{a}_i \boldsymbol{a}_i^T \mathbb{I}_{\{|y_i| \le \alpha_y \lambda_0\}}. \tag{16}$$

Gradient loop: for $t = 0 : T-1$ do

$$\mathbf{z}^{(t+1)} = \mathbf{z}^{(t)} - \frac{\mu}{\|\mathbf{z}^{(0)}\|} \cdot \frac{1}{m}\sum_{i=1}^{m}\left(\mathbf{a}_i^T \mathbf{z}^{(t)} - y_i \cdot \frac{\mathbf{a}_i^T \mathbf{z}^{(t)}}{|\mathbf{a}_i^T \mathbf{z}^{(t)}|}\right)\mathbf{a}_i \mathbb{I}_{\mathcal{E}_i^t}, \tag{17}$$

where

$$\mathcal{E}_i^t := \left\{ i \middle| \left| y_i - |\mathbf{a}_i^T \mathbf{z}^{(t)}| \right| \le \alpha_h \cdot \mathsf{med}\left(\left\{ \left| y_i - |\mathbf{a}_i^T \mathbf{z}^{(t)}| \right| \right\}_{i=1}^m \right) \right\}.$$

Output $\mathbf{z}_T$.

4.2 *Performance Guarantees*

We characterize the performance guarantees of median-TGD for robust phase retrieval. We first show that if initialized in a basin of attraction, median-TGD converges to the ground truth at a linear rate under the Gaussian model with an order of $\mathcal{O}(n \log n)$ measurements, even when a constant fraction of measurements are arbitrarily corrupted, which is order-wise optimal. In addition, the truncated spectral method ensures an initialization in the basin of attraction with an order of $\mathcal{O}(n)$ measurements even when a constant fraction of measurements are arbitrarily corrupted. Furthermore, the stability of median-TGD against additional dense bounded noise is also established.

Theorem 1 summarizes the performance guarantee of median-TGD in Algorithm 2 for phase retrieval in the presence of both sparse arbitrary outliers and dense bounded noise when initialized around a proper neighborhood around the ground truth. The proof can be found in [9].

Theorem 3 *Consider the phase retrieval problem given in* (12)*. Suppose that the initialization* $\mathbf{z}^{(0)}$ *satisfies*

$$\operatorname{dist}\left(\mathbf{z}^{(0)}, \mathbf{x}\right) \le \frac{1}{12}\|\mathbf{x}\|.$$

There exist constants $\mu_0, s_0 > 0$*,* $0 < \rho < 1$ *and* $c_0, c_1, c_2 > 0$ *such that if* $m \ge c_0 n \log n$*,* $s < s_0$*,* $\mu \le \mu_0$*, then with probability at least* $1 - c_1 \exp(-c_2 m)$*, median-TGD yields*

$$\operatorname{dist}(\mathbf{z}^{(t)}, \mathbf{x}) \lesssim \|\mathbf{w}\|_\infty + (1-\rho)^t \|\mathbf{x}\|, \quad \forall t \in \mathbb{N},$$

simultaneously for all $\mathbf{x} \in \mathbb{R}^n \backslash \{\mathbf{0}\}$.

Theorem 4 provides that the truncated spectral method provides an initialization in the basin of attraction with high probability.

Theorem 4 *Fix* $\delta > 0$ *and* $\mathbf{x} \in \mathbb{R}^n$*, and consider the model given by* (12)*. Suppose that* $\|\mathbf{w}\|_\infty \le c_w \|\mathbf{x}\|$ *for some sufficiently small constant* $c_w > 0$ *and that* $\|\|_0 \le sm$ *for some sufficiently small constant* s*. With probability at least* $1 - \exp(-\Omega(m))$*, the initialization given by the median-truncated spectral method obeys*

$$dist(\mathbf{z}^{(0)}, \mathbf{x}) \le \delta \|\mathbf{x}\|,$$

provided that $m > c_0 n$ *for some constant* $c_0 > 0$.

Theorem 3, together with Theorem 4, indicates that median-TGD admits exact recovery for *all* signals in the presence of only sparse outliers with arbitrary magnitudes even when the number of outliers scales linearly with the number of measurements, as long as the sample complexity satisfies $m \gtrsim n \log n$. Moreover, median-TGD converges at a linear rate using a constant step size, with per-iteration cost $\mathcal{O}(mn)$. To reach ϵ-accuracy, i.e., $\operatorname{dist}(\mathbf{z}^{(t)}, \mathbf{x}) \le \epsilon$, only $\mathcal{O}(\log 1/\epsilon)$ iterations are needed, yielding the total computational cost as $\mathcal{O}(mn \log 1/\epsilon)$, which is highly efficient. With both sparse arbitrary outliers and dense bounded noises, Theorem 3 implies that median-TGD achieves the same convergence rate and the same level of estimation error as the model with only bounded noise. Moreover, it can be seen that applying median-TGD does not require the knowledge of the existence of outliers. When there do exist outliers, median-TGD achieves almost the same performance *as if outliers do not exist*.

5 Highlights of Theoretical Analysis

Broadly speaking, the theoretical analysis of median-TGD for both problems follow the same roadmap. The crux is to use the statistical properties of the median to show that the median-truncated gradients satisfy the so-called *Regularity Condition* (RC) [3], which guarantees the linear convergence of the update, provided the initialization provably lands in a small neighborhood of the true signal. We first develop a few statistical properties of median that will be useful throughout our analysis in Sect. 5.1. Section 5.2 explains the RC and how it leads to geometric convergence.

5.1 Useful Properties of Median

The sample median, and the order statistics, possesses a few useful properties that we list here for reference [22]. To begin, we define below the quantile function of a population distribution and its corresponding sample version.

Definition 1 (*Generalized quantile function* [22]) Let $0 < \tau < 1$. For a cumulative distribution function (CDF) $F(x)$, the generalized quantile function is defined as

$$F^{-1}(\tau) = \inf\{x \in \mathbb{R} : F(x) \geq \tau\}.$$

For simplicity, denote $\theta_\tau(F) = F^{-1}(\tau)$ as the τ-quantile of F. Moreover, for a sample collection $\mathbf{y} = \{y_i\}_{i=1}^m$, the sample τ-quantile $\theta_\tau(\mathbf{y})$ means $\theta_\tau(\hat{F})$, where $\hat{F}$ is the empirical distribution of the samples $\mathbf{y}$. Specifically, $\mathrm{med}(\mathbf{y}) = \theta_{1/2}(\mathbf{y})$.

Lemma 1 shows that as long as the sample size is large enough, the sample quantile concentrates around the population quantile.

Lemma 1 *Suppose $F(\cdot)$ is cumulative distribution function (i.e., nondecreasing and right-continuous) with continuous density function $f(\cdot)$. Assume the samples $\{X_i\}_{i=1}^m$ are i.i.d. drawn from f. Let $0 < p < 1$. If there exist lower and upper bounds l, L such that $l < f(\theta) < L$ for all θ in $\{\theta : |\theta - \theta_p| \leq \epsilon\}$, then*

$$\left|\theta_p(\{X_i\}_{i=1}^m) - \theta_p(F)\right| < \epsilon$$

holds with probability at least $1 - 2\exp(-2m\epsilon^2 l^2)$.

Lemma 2 bounds the distance between the median of two sequences.

Lemma 2 *Given a vector $\mathbf{X} = (X_1, X_2, ..., X_n)$, where we order the entries in a nondecreasing manner $X_{(1)} \leq X_{(2)} \leq ... \leq X_{(n-1)} \leq X_{(n)}$. Given another vector $\mathbf{Y} = (Y_1, Y_2, ..., Y_n)$, then*

$$|X_{(k)} - Y_{(k)}| \leq \|\mathbf{X} - \mathbf{Y}\|_\infty$$

holds for all $k = 1, ..., n$.

As an example, consider two sequences $\mathbf{X} = (1, 3, 5, -6, 8)$, and $\mathbf{Y} = (2, 1, -9, 5, 3)$, we have the ordered sequences, respectively, as

$$\begin{aligned}\widetilde{\mathbf{X}} &= (-6, 1, 3, 5, 8),\\ \widetilde{\mathbf{Y}} &= (-9, 1, 2, 3, 5).\end{aligned}$$

It is easy to verify that $\max_{1\leq k\leq 5}|X_{(k)} - Y_{(k)}| = 3 \leq \|\mathbf{X} - \mathbf{Y}\|_\infty = 14$.

Lemma 3, as a key robustness property of median, suggests that in the presence of outliers, one can bound the sample median from both sides by neighboring quantiles of the corresponding clean samples.

Lemma 3 *Consider clean samples $\{\tilde{X}_i\}_{i=1}^m$. If a fraction s ($s < \frac{1}{2}$) of them are corrupted by outliers, one obtains* contaminated samples *$\{X_i\}_{i=1}^m$ which contain sm corrupted samples and $(1-s)m$ clean samples. Then for a quantile p such that $s < p < 1-s$, we have*

$$\theta_{p-s}(\{\tilde{X}_i\}) \leq \theta_p(\{X_i\}) \leq \theta_{p+s}(\{\tilde{X}_i\}).$$

5.2 Regularity Condition

Once the initialization is guaranteed to be within a small neighborhood of the ground truth, we only need to show that the truncated gradients (8) and (15) satisfy the *Regularity Condition* (RC) [3, 23], which guarantees the geometric convergence of median-TGD once the initialization lands into this neighborhood. For conciseness, we write the definition of RC when $\mathbf{z}$ is a vector, and it is straightforward to extend it to the matrix case.

Definition 2 The gradient $\nabla\ell(\mathbf{z})$ is said to satisfy the Regularity Condition $\mathsf{RC}(\mu, \lambda, c)$ if

$$\langle \nabla\ell(\mathbf{z}), \mathbf{z} - \mathbf{x}\rangle \geq \frac{\mu}{2}\|\nabla\ell(\mathbf{z})\|^2 + \frac{\lambda}{2}\|\mathbf{z} - \mathbf{x}\|^2 \tag{16}$$

for all $\mathbf{z}$ obeying $\|\mathbf{z} - \mathbf{x}\| \leq c\|\mathbf{x}\|$.

The above RC guarantees that the gradient descent update $\mathbf{z}^{(t+1)} = \mathbf{z}^{(t)} - \mu\nabla\ell(\mathbf{z})$ converges to the true signal $\mathbf{x}$ geometrically [23] if $\mu\lambda < 1$. This is due to the following argument:

$$\begin{aligned}\mathrm{dist}^2(\mathbf{z} - \mu\nabla\ell(\mathbf{z}), \mathbf{x}) &\leq \|\mathbf{z} - \mu\nabla\ell(\mathbf{z}) - \mathbf{x}\|^2\\ &= \|\mathbf{z} - \mathbf{x}\|^2 + \|\mu\nabla\ell(\mathbf{z})\|^2 - 2\mu\langle \mathbf{z} - \mathbf{x}, \nabla\ell(\mathbf{z})\rangle\\ &\leq \|\mathbf{z} - \mathbf{x}\|^2 + \|\mu\nabla\ell(\mathbf{z})\|^2 - \mu^2\|\nabla\ell(\mathbf{z})\|^2 - \mu\lambda\|\mathbf{z} - \mathbf{x}\|^2\\ &= (1 - \mu\lambda)\,\mathrm{dist}^2(\mathbf{z}, \mathbf{x}).\end{aligned}$$

Therefore, it boils down to establish that RC holds with high probability for the truncated gradients (8) and (15). However, the analysis of median-TGD is more involved due to the truncation procedure in the gradient descent updates. In particular, certain restricted isometry properties (RIP) of the sample median for the class of signals of interest need to be established, which can be thought as an extension of the RIP for the sample mean in compressed sensing literature [11, 24]. We remark that such a result might be of independent interest, and its establishment is nontrivial due to the nonlinear character of the median operation. We refer interested readers to [8, 9] for details.

6 Numerical Experiments

In this section, we provide several numerical experiments to evaluate the performance of the proposed median-TGD algorithms for robust low-rank matrix recovery and robust phase retrieval, respectively. In particular, we examine the robustness of median-TGD with respect to both sparse outliers and dense noise.

6.1 Median-TGD for Low-Rank Matrix Recovery

We first examine the performance of median-TGD, summarized in Algorithm 1, for robust low-rank matrix recovery. As mentioned earlier, for the initialization step, empirically we observe it is not necessary to split the samples into two parts, and hence the matrix in (9) is instead changed to

$$\mathbf{Y} = \frac{1}{m}\sum_{i=1}^{m} y_i \mathbf{A}_i \mathbb{I}_{\{|y_i| \le \alpha_y \cdot \mathrm{med}(|\mathbf{y}|)\}}.$$

We randomly generate the ground truth as a rank r matrix as $\mathbf{M} = \mathbf{X}\mathbf{X}^T$, where $\mathbf{X} \in \mathbb{R}^{n\times r}$ is composed of i.i.d. standard Gaussian random variables. The ith sensing matrix $\mathbf{A}_i$ is generated as $\mathbf{A}_i = (\mathbf{B}_i + \mathbf{B}_i)/\sqrt{2}$, where $\mathbf{B}_i \in \mathbb{R}^{n\times n}$ consists of i.i.d. standard Gaussian random variables, $i = 1, \ldots, m$. The outliers are i.i.d. randomly generated following $10^2 \|\mathbf{M}\|_F \cdot \mathcal{N}(0, 1)$. Moreover, we set $\alpha_y = 12$ and $\alpha_h = 6$, and pick a constant step size $\mu_t = 0.4$. In all experiments, the maximum number of iterations is set as $T = 10^3$. Denote the solution to the algorithm under examination by $\hat{\mathbf{X}}$, and then the normalized estimate error is defined as $\left\|\hat{\mathbf{X}}\hat{\mathbf{X}}^T - \mathbf{M}\right\|_F / \|\mathbf{M}\|_F$. A trial is deemed a success if the normalized estimate error is less than 10^{-6}.

Fix $n = 150$, $r = 5$, and the percentage of outliers as $s = 5\%$. Figure 1 shows the success rates of median-TGD, averaged over 20 trials, with respect to the number of measurements. As a comparison, Fig. 1 also shows the success rates of vanilla-GD [18] in the same setting except for using outlier-free measurements. It can be observed that median-TGD provides comparable performance to that of vanilla-GD

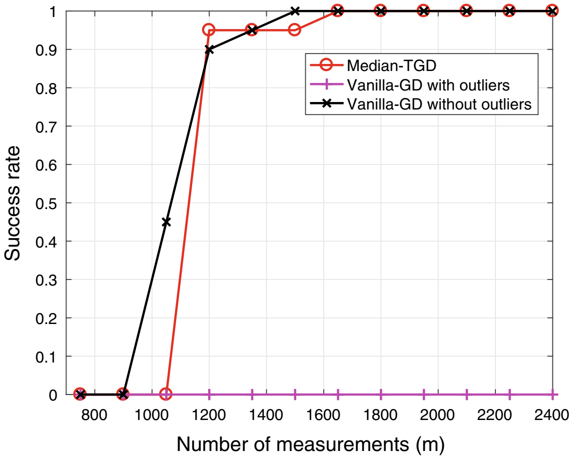


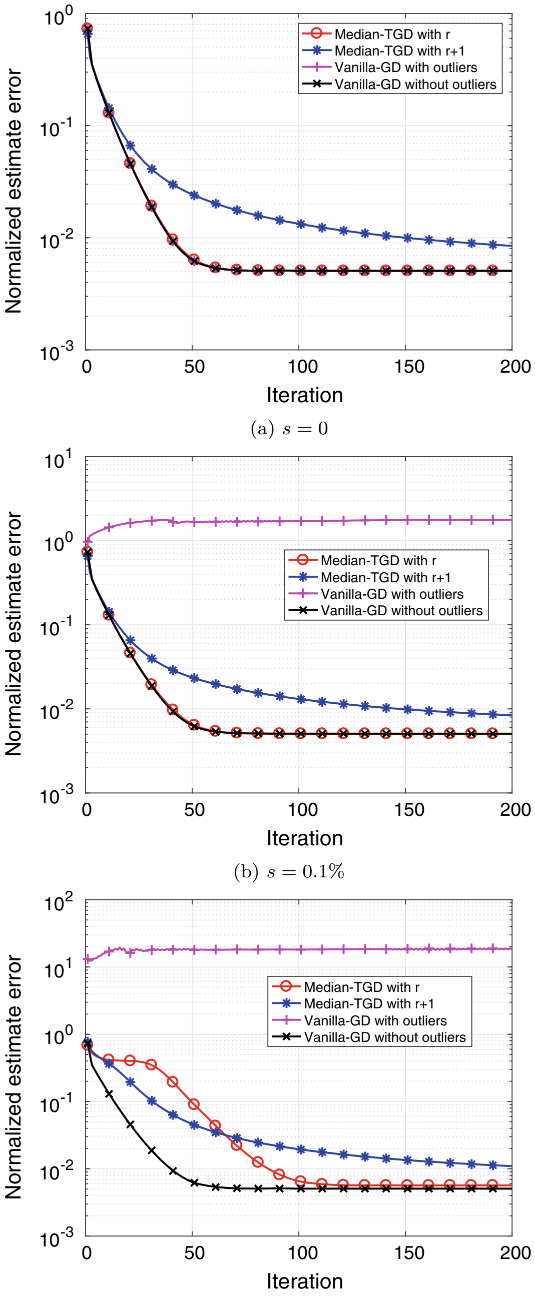
Fig. 1 The success rate of median-TGD for low-rank matrix recovery with respect to the number of measurements, when 5% of measurements are corrupted by outliers, which is similar to that of vanilla-GD using outlier-free measurements. Here, $n = 150$, and $r = 5$

under outlier-free measurements, whose performance dramatically degrades when the measurements are corrupted by outliers. Therefore, median-TGD can deal with outliers in a much more robust manner.

Furthermore, we examine the performance of median-TGD when the measurements are contaminated by both sparse outliers and dense noise. Specifically, the dense noise is generated with i.i.d. random entries following $0.05\sigma_r(\mathbf{M}) \cdot \mathcal{U}[-1, 1]$, and the sparse outliers are generated in the same manner as in Fig. 1. We examine the performance of median-TGD with the true rank r and an inaccurate rank $r + 1$ using outlier-corrupted measurements, and vanilla-GD with true rank r using outlier-corrupted measurements and outlier-free measurements. Figure 2 shows the normalized estimate error with respect to the iteration count, when the percentage of outliers is, respectively, $s = 0$, 0.1% and 10%. In the outlier-free scenario, both algorithms work well and have comparable convergence rates. However, even with a few outliers, vanilla-GD suffers from a dramatic performance degradation, as shown in Fig. 2b. On the other hand, median-TGD shows robust performance against outliers and can still converge to an accurate estimate even with a large fraction of outliers, as shown in Fig. 2c. Finally, the performance of median-TGD is stable to misspecified rank information as long as an upper bound of the truth rank is provided.

6.2 *Median-TGD for Phase Retrieval*

We now examine the performance of median-TGD, summarized in Algorithm 2, for robust phase retrieval. The ground truth is generated as $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{n\times n})$, and the measurement vectors are i.i.d. randomly generated as $\mathbf{a}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{n\times n})$, for $i = 1, \ldots, m$. The outliers are i.i.d. randomly generated from a uniform distribution $\mathcal{U}[0, 10^3 \, \|\mathbf{x}\|]$. In all experiments, a fixed number of iterations is set as $T = 10^3$. Denote the solution



(a) $s = 0$

(b) $s = 0.1\%$

(c) $s = 10\%$

Fig. 2 Normalized estimate error with respect to the iteration count using median-TGD and vanilla-GD in different outlier-corruption scenarios for low-rank matrix recovery, when $n = 150$, $r = 5$, and $m = 1800$

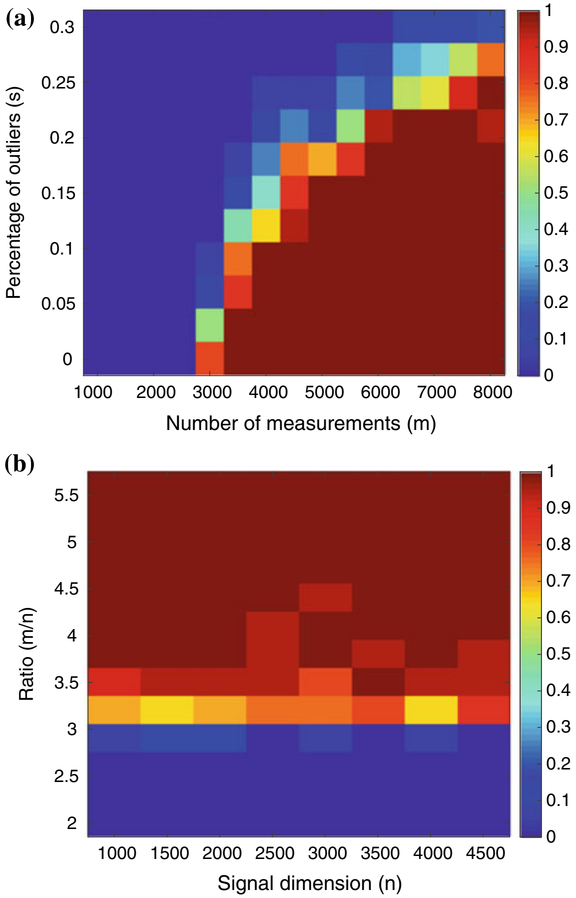


Fig. 3 Success rates of median-TGD for robust phase retrieval. **a** Success rates with respect to the number of measurements m and the percent of outliers s, when the signal dimension $n = 1000$. **b** Success rates with respect to the signal dimension n and the ratio of the number of measurements to the signal dimension m/n, when $s = 5\%$ of measurements are corrupted by outliers

to the algorithm under examination as $\hat{\mathbf{x}}$, and then the normalized estimate error is defined as $\text{dist}(\hat{\mathbf{x}}, \mathbf{x}) / \|\mathbf{x}\|$. A trial is declared successful if the normalized estimate error is less than 10^{-8}.

Figure 3a shows the success rates of median-TGD, averaged over 20 trials, with respect to the number of measurements, and the percentage of outliers when the signal dimension is fixed as $n = 1000$. The performance of median-TGD degenerates smoothly with the increase in the percentage of outliers. Under the same setup as Fig. 3a, b shows the success rate of median-TGD with respect to the signal dimension n and the oversampling ratio m/n, when 5% of measurements are corrupted by outliers. It can be seen that exact recovery can be achieved as soon as the oversampling ratio is above 4.5.

We next examine the performance of median-TGD in the presence of both sparse outliers and dense noise. In particular, we consider the case when the measurements are corrupted by both outliers and the Poisson noise, modeling photon detection in

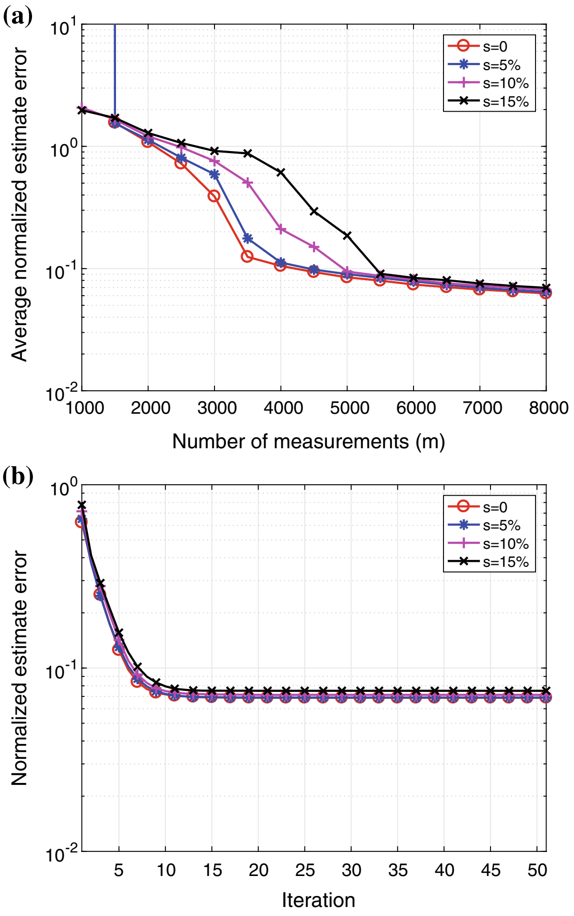
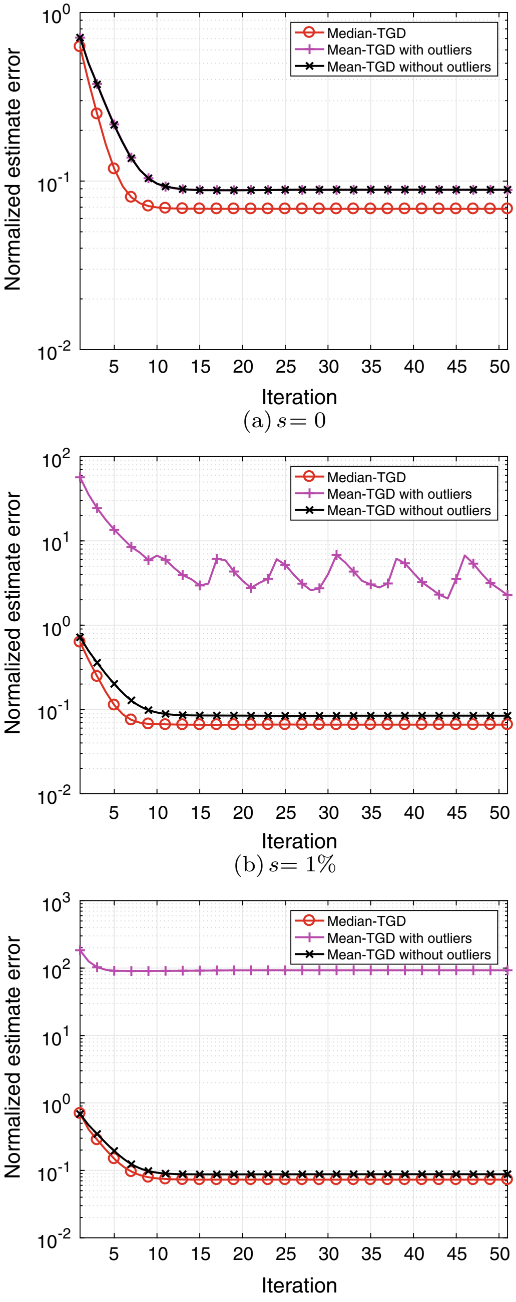


Fig. 4 Performance of median-TGD for robust phase retrieval with sparse outliers and Poisson noise. **a** Averaged normalized estimate error with respect to the number of measurements in different outlier-corruption scenarios, when $n = 1000$. **b** Normalized estimate error with respect to the iteration count with $m = 7000$ and $n = 1000$

optical imaging applications, where each noise-contaminated measurement is i.i.d. randomly generated as $y_i \sim \text{Poisson}(\left|\mathbf{a}_i^T \mathbf{x}\right|)$, for $i = 1, \ldots, m$. The measurements are then further corrupted by outliers that are i.i.d. randomly generated following $\mathcal{U}[0, 10^3 \|\mathbf{x}\|]$, rounded to the nearest integers. Figure 4a depicts the averaged normalized estimate errors over 20 trials with respect to the number of measurements in various outlier-corruption scenarios, where $n = 1000$, and the percent of outliers is set as $s = 0$, $s = 5\%$, $s = 10\%$, and $s = 15\%$, respectively. The performance of TGD is robust against a large fraction of outliers, and median-TGD leads to stable recovery of the underlying signal. The convergence rates of median-TGD are further depicted in Fig. 4b under the same setting of Fig. 4a with $m = 7000$. Finally, under the same setting of Fig. 4b, we compare the performance of median-TGD and mean-TGD [23] using outlier-corrupted measurements and outlier-free measurements. Figure 5 depicts the normalized estimate error with respect to the iteration count, when the percentage of outliers is, respectively, $s = 0$, $s = 1\%$, and $s = 10\%$. In the



(a) $s = 0$

(b) $s = 1\%$

(c) $s = 10\%$

Fig. 5 Normalized estimate error with respect to the iteration count using median-TGD and mean-TGD in different outlier-corruption scenarios for phase retrieval, when $m = 7000$ and $n = 1000$

outlier-free scenario, both algorithms work well and achieve comparable convergence rates. Nevertheless, even with a few outliers, mean-TGD suffers from a dramatic performance degradation, as shown in Fig. 5b, while median-TGD is able to still robustly work and converge to an accurate estimate even with a large fraction of outliers, as shown in Fig. 5c. It can be seen that median-TGD under both outliers and Poisson noise has almost the same accuracy as, if not better than, mean-TGD under only the Poisson noise.

7 Related Works

Developing nonconvex methods with provable global convergence guarantees has attracted intensive research interest recently [25]. A partial list of these studies include phase retrieval [3, 21, 23, 26–31], matrix completion [29, 32–39], low-rank matrix recovery [18, 40–46], robust PCA [47, 48], robust tensor decomposition [49], dictionary learning [50, 51], community detection [52], phase synchronization [53], blind deconvolution [5, 29, 54], and joint alignment [55], to name a few. The median-TGD algorithm provides a new instance in this list that emphasizes robust high-dimensional signal estimation with possibly adversarial outliers.

The concept of median has been adopted in machine learning in various contexts, for example, K-median clustering [56] and resilient data aggregation for sensor networks [57]. The median-TGD algorithm presented here further extends the applications of median to robust high-dimensional estimation problems with theoretical guarantees. Another popular approach in robust estimation is to use the trimmed mean [6], which has found success in robustifying sparse regression [58], subspace clustering [59], etc. However, using the trimmed mean requires knowledge of an upper bound on the number of outliers, whereas median does not require such information. Very recently, geometric median is also adopted for robust empirical risk minimization [60–62].

For the phase retrieval problem, median-TGD is closely related to the truncated Wirtinger flow (TWF) algorithm [23], which is also a truncated gradient descent algorithm for phase retrieval. However, the truncation rule in TWF is based on the sample mean, which is very sensitive to outliers. In [63–65], the problem of phase retrieval with outliers is investigated, but the algorithms therein either lack performance guarantees or are computationally too expensive. For the low-rank matrix recovery problem, median-TGD is closely related to the outlier-free models studied in [18, 42] as a robust counterpart.

To handle outliers, existing convex optimization approaches are often based on sparse and low-rank decompositions, using semidefinite programming [63, 66]. However, the computational cost is very expensive. It is worth mentioning that other nonconvex approaches for robust low-rank matrix completion have been presented in [48, 67, 68], where the goal is to separate a low-rank matrix and sparse outliers from a small number of direct or linear measurements of their sum. The approaches typically use thresholding-based truncation for outlier removal and projected gradient descent for low-rank matrix recovery, which are somewhat similar to median-TGD albeit with different truncation rules. However, this line of work requires stronger assumptions on the outliers such as spreadness conditions.

Central to the proof of the theoretical performance guarantees is a regularity condition [3] that the proposed median-truncated gradient satisfies, which is a sufficient condition for establishing the linear convergence to the ground truth, and has been employed successfully in the analysis of phase retrieval [3, 21–23], blind deconvolution [5], and low-rank matrix recovery [18, 42, 43] in the recent literature, to name a few.

8 Conclusion

In this chapter, we presented median-TGD as a general technique to improve the robustness of vanilla gradient descent for nonconvex statistical estimation in the presence of outliers. The effectiveness of median-TGD is probably guaranteed by theoretical analysis, and validated through numerical experiments, for two important case studies—low-rank matrix recovery and phase retrieval. We expect median-TGD might be useful to other nonconvex estimation problems such as blind deconvolution and matrix completion; however, new techniques might be needed to develop its theoretical performance, since much less randomness is present in the measurement process of those problems.

Acknowledgements The work of Y. Chi and Y. Li is supported in part by AFOSR under the grant FA9550-15-1-0205, by ONR under the grant N00014-18-1-2142, by ARO under the grant W911NF-18-1-0303, and by NSF under the grants CAREER ECCS-1818571 and CCF-1806154. The work of Y. Liang is supported in part by NSF under the grants CCF-1761506 and ECCS-1818904.

References

1. P. Auer, M. Herbster, M.K. Warmuth, Exponentially many local minima for single neurons, in *Advances in neural information processing systems* (1996), pp. 316–322
2. R. Sun, Z.-Q. Luo, Guaranteed matrix completion via non-convex factorization. IEEE Trans. Inf. Theory **62**(11), 6535–6579 (2016)
3. E.J. Candès, X. Li, M. Soltanolkotabi, Phase retrieval via wirtinger flow: theory and algorithms. IEEE Trans. Inf. Theory **61**(4), 1985–2007 (2015)
4. J. Sun, Q. Qu, J. Wright, Complete dictionary recovery over the sphere i: overview and the geometric picture. IEEE Trans. Inf. Theory **63**(2), 853–884 (2017)
5. X. Li, S. Ling, T. Strohmer, K. Wei, Rapid, robust, and reliable blind deconvolution via nonconvex optimization. Appl. Comput. Harmon. Anal. (2018)
6. P.J. Huber, *Robust Statistics* (Springer, 2011)
7. Y. Li, Y. Chi, H. Zhang, Y. Liang, Non-convex low-rank matrix recovery from corrupted random linear measurements, in *2017 International Conference on Sampling Theory and Applications (SampTA)* (2017)

8. Y. Li, Y. Chi, H. Zhang, Y. Liang, Nonconvex low-rank matrix recovery with arbitrary outliers via median-truncated gradient descent. arXiv:1709.08114 (2017)
9. H. Zhang, Y. Chi, Y. Liang, Median-truncated nonconvex approach for phase retrieval with outliers. IEEE Trans. Inf. Theory **64**(11), 7287–7310 (2018)
10. R.J. Tibshirani, Fast computation of the median by successive binning. arXiv:0806.3301 (2008)
11. B. Recht, M. Fazel, P.A. Parrilo, Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. SIAM Rev. **52**(3), 471–501 (2010)
12. D. Gross, Recovering low-rank matrices from few coefficients in any basis. IEEE Trans. Inf. Theory **57**(3), 1548–1566 (2011)
13. S. Negahban, M.J. Wainwright, Estimation of (near) low-rank matrices with noise and high-dimensional scaling. Ann. Stat. **39**(2), 1069–1097 (2011)
14. E. Candes, B. Recht, Exact matrix completion via convex optimization. Commun. ACM **55**(6), 111–119 (2012)
15. Y. Chen, Y. Chi, Robust spectral compressed sensing via structured matrix completion. IEEE Trans. Inf. Theory **60**(10), 6576–6601 (2014)
16. Y. Chen, Y. Chi, A. Goldsmith, Exact and stable covariance estimation from quadratic sampling via convex programming. IEEE Trans. Inf. Theory **61**(7), 4034–4059 (2015)
17. Y. Chen, Y. Chi, Harnessing structures in big data via guaranteed low-rank matrix estimation: recent theory and fast algorithms via convex and nonconvex optimization. IEEE Signal Process. Mag. **35**(4), 14–31 (2018)
18. S. Tu, R. Boczar, M. Simchowitz, M. Soltanolkotabi, B. Recht, Low-rank solutions of linear matrix equations via procrustes flow, in *Proceedings of the 33rd International Conference on International Conference on Machine Learning (ICML)* (2016), pp. 964–973
19. J. Drenth, *X-Ray Crystallography* (Wiley Online Library, 2007)
20. J.R. Fienup, Phase retrieval algorithms: a comparison. Appl. Opt. **21**(15), 2758–2769 (1982)
21. H. Zhang, Y. Zhou, Y. Liang, Y. Chi, A nonconvex approach for phase retrieval: reshaped wirtinger flow and incremental algorithms. J. Mach. Learn. Res. **18**(141), 1–35 (2017)
22. H. Zhang, Y. Chi, Y. Liang, Provable non-convex phase retrieval with outliers: Median truncated wirtinger flow, in *International Conference on Machine Learning* (2016), pp. 1022–1031
23. Y. Chen, E. Candes, Solving random quadratic systems of equations is nearly as easy as solving linear systems, in *Advances in Neural Information Processing Systems (NIPS)* (2015)
24. E.J. Candès, Y. Plan, Tight oracle inequalities for low-rank matrix recovery from a minimal number of noisy random measurements. IEEE Trans. Inf. Theory **57**(4), 2342–2359 (2011)
25. Y. Chi, Y. M. Lu, Y. Chen, Nonconvex optimization meets low-rank matrix factorization: an overview. arXiv:1809.09573 (2018)
26. P. Netrapalli, P. Jain, S. Sanghavi, Phase retrieval using alternating minimization, *Advances in Neural Information Processing Systems (NIPS)* (2013)
27. G. Wang, G.B. Giannakis, Y.C. Eldar, Solving systems of random quadratic equations via truncated amplitude flow. IEEE Trans. Inf. Theory **64**(2), 773–794 (2018)
28. J. Sun, Q. Qu, J. Wright, A geometric analysis of phase retrieval. Found. Comput. Math. **18**(5), 1131–1198 (2018)
29. C. Ma, K. Wang, Y. Chi, Y. Chen, Implicit regularization in nonconvex statistical estimation: gradient descent converges linearly for phase retrieval, matrix completion and blind deconvolution. arXiv:1711.10467 (2017)
30. Y. Li, C. Ma, Y. Chen, Y. Chi, Nonconvex matrix factorization from rank-one measurements. arXiv:1802.06286 (2018)
31. Y. Chen, Y. Chi, J. Fan, C. Ma, Gradient descent with random initialization: fast global convergence for nonconvex phase retrieval. arXiv:1803.07726 (2018)
32. R.H. Keshavan, A. Montanari, S. Oh, Matrix completion from a few entries. IEEE Trans. Inf. Theory **56**(6), 2980–2998 (2010)
33. P. Jain, P. Netrapalli, S. Sanghavi, Low-rank matrix completion using alternating minimization, in *Proceedings of the Forty-Fifth Annual ACM Symposium on Theory of Computing* (2013), pp. 665–674

34. R. Sun, Z.-Q. Luo, Guaranteed matrix completion via nonconvex factorization, in *IEEE 56th Annual Symposium on Foundations of Computer Science (FOCS)* (2015), pp. 270–289
35. M. Hardt, Understanding alternating minimization for matrix completion, in *IEEE 55th Annual Symposium on Foundations of Computer Science (FOCS)* (2014), pp. 651–660
36. C. De Sa, C. Re, K. Olukotun, Global convergence of stochastic gradient descent for some non-convex matrix problems, in *International Conference on Machine Learning* (2015), pp. 2332–2341
37. Q. Zheng, J. Lafferty, Convergence analysis for rectangular matrix completion using burer-monteiro factorization and gradient descent. arXiv:1605.07051 (2016)
38. C. Jin, S. M. Kakade, P. Netrapalli, Provable efficient online matrix completion via non-convex stochastic gradient descent, in *Advances in Neural Information Processing Systems* (2016), pp. 4520–4528
39. R. Ge, J. D. Lee, T. Ma, Matrix completion has no spurious local minimum, in *Advances in Neural Information Processing Systems (NIPS)* (2016), pp. 2973–2981
40. S. Bhojanapalli, B. Neyshabur, N. Srebro, Global optimality of local search for low rank matrix recovery, in *Advances in Neural Information Processing Systems* (2016), pp. 3873–3881
41. Y. Chen, M.J. Wainwright, Fast low-rank estimation by projected gradient descent: general statistical and algorithmic guarantees. arXiv:1509.03025 (2015)
42. Q. Zheng, J. Lafferty, A convergent gradient descent algorithm for rank minimization and semidefinite programming from random linear measurements, in *Advances in Neural Information Processing Systems (NIPS)* (2015)
43. D. Park, A. Kyrillidis, C. Caramanis, S. Sanghavi, Finding low-rank solutions via nonconvex matrix factorization, efficiently and provably. SIAM J. Imaging Sci. **11**(4), 2165–2204 (2018)
44. K. Wei, J.-F. Cai, T.F. Chan, S. Leung, Guarantees of riemannian optimization for low rank matrix recovery. SIAM J. Matrix Anal. Appl. **37**(3), 1198–1222 (2016)
45. Q. Li, G. Tang, The nonconvex geometry of low-rank matrix optimizations with general objective functions, in *2017 IEEE Global Conference on Signal and Information Processing (GlobalSIP)* (IEEE, 2017), pp. 1235–1239
46. X. Li, J. Haupt, J. Lu, Z. Wang, R. Arora, H. Liu, T. Zhao, Symmetry, saddle points, and global optimization landscape of nonconvex matrix factorization, in *Information Theory and Applications Workshop (ITA)* (IEEE 2018), pp. 1–9
47. P. Netrapalli, U. Niranjan, S. Sanghavi, A. Anandkumar, P. Jain, Non-convex robust PCA, in *Advances in Neural Information Processing Systems (NIPS)* (2014)
48. X. Yi, D. Park, Y. Chen, C. Caramanis, Fast algorithms for robust PCA via gradient descent, in *Advances in Neural Information Processing Systems* (2016), pp. 4152–4160
49. A. Anandkumar, P. Jain, Y. Shi, U.N. Niranjan, Tensor vs. matrix methods: Robust tensor decomposition under block sparse perturbations, in *Artificial Intelligence and Statistics* (2016), pp. 268–276
50. S. Arora, R. Ge, T. Ma, A. Moitra, Simple, efficient, and neural algorithms for sparse coding, in *Conference on Learning Theory* (2015), pp. 113–149
51. J. Sun, Q. Qu, J. Wright, Complete dictionary recovery using nonconvex optimization, in *Proceedings of the 32nd International Conference on Machine Learning (ICML)* (2015)
52. A. S. Bandeira, N. Boumal, V. Voroninski, On the low-rank approach for semidefinite programs arising in synchronization and community detection, in *29th Annual Conference on Learning Theory* (2016)
53. N. Boumal, Nonconvex phase synchronization. SIAM J. Optim. **26**(4), 2355–2377 (2016)
54. K. Lee, Y. Li, M. Junge, Y. Bresler, Blind recovery of sparse signals from subsampled convolution. IEEE Trans. Inf. Theory **63**(2), 802–821 (2017)
55. Y. Chen, E.J. Candès, The projected power method: an efficient algorithm for joint alignment from pairwise differences. Commun. Pure Appl. Math. **71**(8), 1648–1714 (2018)
56. K. Chen, On k-median clustering in high dimensions, in *Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithm* (2006)
57. D. Wagner, Resilient aggregation in sensor networks, in *Proceedings of the 2nd ACM Workshop on Security of Ad Hoc and Sensor Networks* (ACM, 2004), pp. 78–87

58. Y. Chen, C. Caramanis, S. Mannor, Robust sparse regression under adversarial corruption, in *Proceedings of the 30th International Conference on Machine Learning (ICML)* (2013)
59. C. Qu, H. Xu, Subspace clustering with irrelevant features via robust dantzig selector, in *Advances in Neural Information Processing Systems (NIPS)* (2015)
60. A. Prasad, A. S. Suggala, S. Balakrishnan, P. Ravikumar, Robust estimation via robust gradient estimation. arXiv:1802.06485 (2018)
61. D. Yin, Y. Chen, R. Kannan, P. Bartlett, Byzantine-robust distributed learning: towards optimal statistical rates, in *Proceedings of the 35th International Conference on Machine Learning, 10–15 Jul 2018* (2018), pp. 5650–5659
62. Y. Chen, L. Su, J. Xu, Distributed statistical machine learning in adversarial settings: byzantine gradient descent, in *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, vol. 1, no 2 (2017), p. 44
63. Y. Li, Y. Sun, Y. Chi, Low-rank positive semidefinite matrix recovery from corrupted rank-one measurements. IEEE Trans. Signal Process. **65**(2), 397–408 (2017)
64. P. Hand, Phaselift is robust to a constant fraction of arbitrary errors. Applied and Computational Harmonic Analysis **42**(3), 550–562 (2017)
65. D. Weller, A. Pnueli, G. Divon, O. Radzyner, Y. Eldar, J. Fessler, Undersampled phase retrieval with outliers. IEEE Transactions on Computational Imaging **1**(4), 247–258 (2015)
66. J. Wright, A. Ganesh, K. Min, Y. Ma, Compressive principal component pursuit. Information and Inference **2**(1), 32–68 (2013)
67. Y. Cherapanamjeri, K. Gupta, P. Jain, Nearly optimal robust matrix completion, in *Proceedings of the 34th International Conference on Machine Learning* (2017), pp. 797–805
68. X. Zhang, L. Wang, Q. Gu, A unified framework for nonconvex low-rank plus sparse matrix recovery, in *International Conference on Artificial Intelligence and Statistics* (2018), pp. 1097–1107

Reconstruction Methods in THz Single-Pixel Imaging

Martin Burger, Lea Föcke, Lukas Nickel, Peter Jung and Sven Augustin

Abstract The aim of this paper is to discuss some advanced aspects of image reconstruction in single-pixel cameras, focusing in particular on detectors in the THz regime. We discuss the reconstruction problem from a computational imaging perspective and provide a comparison of the effects of several state-of-the-art regularization techniques. Moreover, we focus on some advanced aspects arising in practice with THz cameras, which lead to nonlinear reconstruction problems: the calibration of the beam reminiscent of the Retinex problem in imaging and phase recovery problems. Finally, we provide an outlook to future challenges in the area.

1 Introduction

Imaging science has been a strongly evolving field in the past century, with a lot of interesting developments concerning devices, measurement strategies and computational approaches to obtain high-quality images. A current focus concerns imaging from undersampled data in order to allow novel developments toward dynamic and hyperspectral imaging, where time restrictions forbid to acquire full samplings. In

M. Burger (✉) · L. Föcke
Friedrich-Alexander Universität Erlangen-Nürnberg (FAU Erlangen-Nürnberg), Erlangen, Germany
e-mail: martin.burger@fau.de

L. Föcke
e-mail: lea.foecke@fau.de

L. Nickel
Westfälische-Wilhelms Universität Münster (WWU Münster), Münster, Germany
e-mail: lukas.nickel@uni-muenster.de

P. Jung
Technische Universität Berlin (TU Berlin), Berlin, Germany
e-mail: peter.jung@tu-berlin.de

S. Augustin
Humboldt-Universität zu Berlin (HU Berlin), Berlin, Germany
e-mail: sven.augustin@dlr.de

H. Boche et al. (eds.), *Compressed Sensing and Its Applications*,
Applied and Numerical Harmonic Analysis,
https://doi.org/10.1007/978-3-319-73074-5_9

order to compensate for the sampling either physical models (e.g., concerning motion in dynamic imaging, cf. [8]) or a priori knowledge about the images to be reconstructed are used. In applications, where one has sufficient freedom to choose the undersampling patterns, the paradigm of compressed sensing is particularly popular. It is based on minimizing the coherence between measurements (cf. [22]), often achieved by random sampling (cf. [5])

Single-pixel imaging, or more precisely single-detector imaging, is one of the most interesting developments in compressed sensing (cf. [15, 23, 26, 57]). It is based on using a single detector with multiple random masks in order to achieve the desired resolution. The ideal model is to have the mask realize a grid on the detector region with subpixels of the desired image resolution, which are either open or closed with a certain probability. The detector integrates the light passing through the open subpixels in the mask. Note that the image at desired resolution might be acquired by scanning in a deterministic way, in the easiest setting with a single subpixel open at each shot. However, the light intensity obtained from a single subpixel might not be sufficient to obtain a reasonable signal-to-noise ratio and obviously, the mechanical scanning times may strongly exceed the potential times of undersampling with masks having multiple open subpixels.

This idea is particularly relevant in applications, where detectors are expensive or difficult to miniaturize such as imaging in the THz range (cf. [2, 3, 15, 59]), which is our main source of motivation. The random masks are achieved by a spatial light modulator, which may, however, deviate from the ideal setting in practical applications due to the following effects:

- *Beam calibration*: as in many practical imaging approaches, the lighting beam is not homogeneous and needs to be corrected, a problem reminiscent of the classical Retinex problem (cf. [46]).
- *Diffraction*: deviations between the object and image plane may cause the need to consider diffraction and take care of out-of-phase effects, which complicates the inversion problem and
- *Motion*: The object to be imaged may move between consecutive shots, i.e., while changing the masks, which leads to well-known motion blur effects in reconstructions.

In the following, we will discuss such issues in image reconstruction arising in THz single-pixel cameras. We start with the basic modeling of the image formation excluding calibration and further effects in Sect. 2, where we also discuss several regularization models. In Sect. 3, we discuss some computational methods to efficiently solve the reconstruction problem and in particular, compare the results of some state-of-the-art regularization techniques. Then, we discuss the challenges that arise when applying the single-pixel imaging approach in a practical setup with THz detectors and proceed to calibration models for beams in Sect. 5, where we provide a relation to the classical Retinex problem and discuss the particular complications arising due to the combination with the single-pixel camera. In Sect. 6 we discuss the phase reconstruction problem and compare several state-of-the-art approaches for such. Finally, we conclude and discuss further challenges in Sect. 7.

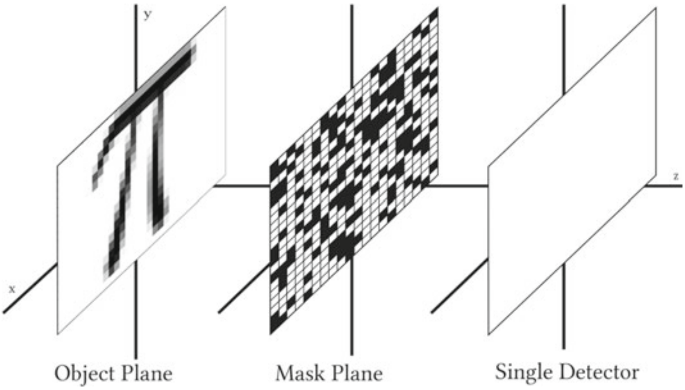


Fig. 1 Sketch of the setup of imaging planes in THz single-pixel imaging (from [50])

2 Compressed Sensing and Reconstruction in Single-Pixel Cameras

In the following, we discuss some basic aspects of the image reconstruction in single-pixel cameras. We are interested in reconstructing a two-dimensional image on the subpixel grid of size $d_1 \times d_2$, i.e., $\mathbf{p} \in \mathbb{R}^n$ where $n = d_1 \cdot d_2$. The simplest model of the image formation process is to have each measurement y_i, $i = 1, \ldots, m$, as the sum of subpixel values p_j for those j corresponding to the open subpixels in the ith mask. This means we can write

$$\mathbf{y} = \mathbf{A}\mathbf{p},$$

with a matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ whose entries are only zeros and ones. Again the nonzero entries in each row of $\mathbf{A}$ correspond to the open subpixels of the respective mask (compare Fig. 1 for a schematic overview of such a setup).

Choosing $m \geq n$ deterministic masks appropriately one could guarantee that $\mathbf{A}$ is invertible and simply solve the linear reconstruction problem. However, in practice, one would like to reconstruct a reasonable image $\mathbf{p}$ from a number of measurements m significantly smaller than n. Single- pixel cameras are hence developed in the paradigm of compressed sensing and the natural way to realize appropriate measurements is to choose the masks randomly. In most systems such as the ones we consider here, each entry of $\mathbf{A}$ is chosen independently as a binary random variable with fixed expectation. Combining such approaches with appropriate prior information in terms of sparsity leads to compressed sensing schemes that can be analyzed rigorously (cf. [17, 44]).

2.1 Compressed Sensing Techniques

A key motivation for compressed sensing comes from the fact that in many cases $d_1 \times d_2$ images are compressible and can be (approximately) sparsely represented as

$\mathbf{p} = \Psi\boldsymbol{\alpha}$ with $\boldsymbol{\alpha} \in \mathbb{R}^n$ and a particular basis $\Psi \in \mathbb{R}^{n\times n}$, e.g., wavelets or overcomplete systems $\Psi \in \mathbb{R}^{n\times N}$ with $N > n$ such as shearlets (cf. [33]). By this, we mean that

$$\|\boldsymbol{\alpha}\|_{\ell_0} = |\{k \,:\, \alpha_k \neq 0\}|,$$

respectively

$$|\{k \,:\, |\alpha_k| \geq \epsilon\}|,$$

for small $\epsilon > 0$, is considerably smaller then the ambient dimension n. Ideally, one would thus solve the problem of minimizing $\|\boldsymbol{\alpha}\|_{\ell_0}$ subject to the linear constraint $\mathbf{A}\Psi\boldsymbol{\alpha} = \mathbf{y}$, which is, however, an NP-hard combinatorial problem.

One of the fundamental results that initiated the field of compressed sensing (cf. [19, 25, 28, 30]) is that under additional assumptions on the map $\mathbf{A}\Psi$ the convex relaxation

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_{\ell_1} \quad \text{s.t.} \quad \mathbf{y} = \mathbf{A}\Psi\boldsymbol{\alpha} \tag{1}$$

recovers exactly (in the noiseless setting) the unknown $\boldsymbol{\alpha}$ yielding the correct image $\mathbf{p}$. A common additional assumption is that the image itself has nonnegative pixel intensities, $\mathbf{p} \geq 0$ such that problem (1) is extended to

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_{\ell_1} \quad \text{s.t.} \quad \mathbf{y} = \mathbf{A}\Psi\boldsymbol{\alpha} \quad \text{and} \quad \Psi\boldsymbol{\alpha} \geq 0. \tag{2}$$

If $\Psi = \text{Id}$ and if the row span of $\mathbf{A}$ intersects the positive orthant, meaning that there exists a vector $\mathbf{t}$ such that $\mathbf{A}^*\mathbf{t} > 0$, the measurement matrix $\mathbf{A}$ itself already assures that the ℓ_1-norm $\|\mathbf{p}\|_{\ell_1}$ of a feasible $\mathbf{p} = \boldsymbol{\alpha}$ in (2) is equal (or close in the noisy case) to $\|\mathbf{p}^0\|_{\ell_1}$ where $\mathbf{p}^0$ is the unknown image to recover. To see this, let us assume exemplary that we find a vector $\mathbf{t}$ such that $\mathbf{A}^*\mathbf{t} = \mathbf{1}$ is the all-one vector and $\mathbf{y} = \mathbf{A}\mathbf{p}^0$. Then

$$\begin{aligned}\|\mathbf{p}\|_{\ell_1} - \|\mathbf{p}^0\|_{\ell_1} &= \langle \mathbf{1}, \mathbf{p} - \mathbf{p}^0\rangle = \langle \mathbf{t}, \mathbf{A}(\mathbf{p} - \mathbf{p}^0)\rangle \\ &= \langle \mathbf{t}, \mathbf{A}\mathbf{p} - \mathbf{y}\rangle \leq \|\mathbf{t}\|_{\ell_2}\|\mathbf{A}\mathbf{p} - \mathbf{y}\|_{\ell_2}\end{aligned}$$

and hence, it is enough to minimize the residual over $\mathbf{p} \geq 0$ and replace (2) by a simple non-negative least squares (NNLS) problem:

$$\min_{\mathbf{p}\geq 0} \|\mathbf{y} - \mathbf{A}\mathbf{p}\|_{\ell_2}^2. \tag{3}$$

Indeed, that under these assumptions ℓ_1-minimization reduces to a feasibility problem has observed already in prior work [9, 60]. In particular, the setting of random binary masks has been investigated in [40, 56] and a considerably (partially-) derandomized result based on orthogonal arrays is discussed in [37]. Although this explains to some extent why nonnegativity and NNLS are very useful in certain imaging problems this does not easily extends to generic dictionaries Ψ.

Hence, coming back to (2), in the case of noisy data knowledge about the expected solution should be included. Hence, one usually rather solves

$$\min_{\boldsymbol{\alpha}} \frac{1}{2}\|\mathbf{y}-\mathbf{A}\Psi\boldsymbol{\alpha}\|_{\ell_2}^2+\lambda\|\boldsymbol{\alpha}\|_{\ell_1} \quad \text{s.t.} \quad \Psi\boldsymbol{\alpha}\geq 0,$$

where $\lambda > 0$ is an appropriate regularization parameter.

2.2 Total Variation Regularization and Related Methods

While simple ℓ_1-regularization is a common approach used for the theory of compressed sensing, for image reconstruction this choice as data fitting term has some drawbacks in practice, e.g., in wavelets the reconstructions may suffer from artifacts due to rectangular structures in their constructions. In more advanced models like shearlets (cf. [42]), the visual quality of reconstructions is improved, but the computational overhead in such large dictionaries may become prohibitive. A much more popular approach in practical image reconstructions are total variation based methods such as minimizing

$$\min_{\mathbf{p}} \|\nabla\mathbf{p}\|_{\ell_1} \quad \text{s.t.} \quad \mathbf{y}=\mathbf{A}\mathbf{p} \tag{4}$$

or the penalty version

$$\min_{\mathbf{p}} \frac{1}{2}\|\mathbf{y}-\mathbf{A}\mathbf{p}\|_{\ell_2}^2+\lambda\|\nabla\mathbf{p}\|_{\ell_1},$$

potentially with additional nonnegativity constraints. Total variation methods in compressed sensing have been investigated recently in further detail (cf. [21, 41, 52–54]).

As in wavelet systems, simple approaches in total variation may suffer from rectangular grid artifacts. This is the case in particular if the straight ℓ_1-norm of $\nabla\mathbf{p}$ is used in the regularization (namely, $\|\nabla\mathbf{p}\|_{\ell_{1,1}}$), which corresponds to an anisotropic total variation. It is well known that such approaches promote rectangular structures aligned with the coordinate axis (cf. [14]), but destroy round edges. An improved version is the isotropic version, which considers $\nabla\mathbf{p}\in\mathbb{R}^{n\times 2}$ (the rows corresponding to partial derivatives) and computes $\|\nabla\mathbf{p}\|_{\ell_{2,1}}$ as total variation. The isotropic total variation promotes round structures that are visually more appealing in natural images, which can be made precise in a continuum limit.

A remaining issue of total variation regularization in applications is the so-called stair-casing phenomenon, which means that piecewise constant structures are promoted too strongly and thus gradual changes in an image are rather approximated by piecewise constants in a stair-like fashion. In order to cure such issues infimal convolutions of total variation and higher order functionals are often considered, the most prominent one being the total generalized variation (TGV) model (cf. [11])

$$\min_{\mathbf{p},\mathbf{w}} \|\nabla\mathbf{p}-\mathbf{w}\|_{\ell_1}+\beta\|\nabla\mathbf{w}\|_{\ell_1} \quad \text{s.t.} \quad \mathbf{y}=\mathbf{A}\mathbf{p}, \tag{5}$$

where $\beta > 0$ is a parameter to be chosen appropriately. Again, in practice, suitable isotropic variants of the ℓ_1-norm are used, and in most approaches only the symmetrized gradient of the vector field $\mathbf{w}$ is used instead of the full gradient.

In the penalized form for noisy data used in practice, the models above can be written in the unified form

$$\min_{\mathbf{p}} \frac{1}{2}\|\mathbf{y} - \mathbf{A}\mathbf{p}\|_{\ell_2}^2 + \lambda R(\Phi\mathbf{p}),$$

with R being an appropriately chosen seminorm and a suitable matrix Φ, e.g., the gradient or $\Phi = \Psi^{-1}$ in the case of using a basis, or $\Phi = \Psi^*$ for analysis sparsity. A common issue in problems of this form that aim to reduce variance is an increase of bias, which, e.g., leads to a loss of contrast and small structures in total variation regularization. In order to improve upon this issue, iterative regularization (cf. [6, 12]) can be carried out, in the simplest case by the Bregman iteration, which computes each iteration $\mathbf{p}^{k+1}$ as the minimizer of

$$\min_{\mathbf{p}} \frac{1}{2}\|\mathbf{y} - \mathbf{A}\mathbf{p}\|_{\ell_2}^2 + \tilde{\lambda}(R(\Phi\mathbf{p}) - \mathbf{q}^k \cdot \mathbf{p})$$

with subgradient $\mathbf{q}^k \in \partial R(\Phi\mathbf{p}^k)$. In this way in each iteration, a suitable distance to the last iterate is penalized instead of the original functional R, which is effectively a distance to zero. The parameter $\tilde{\lambda}$ is chosen much larger than the optimal λ in the above problem, roughly speaking when carrying out K iterations we have $\tilde{\lambda} = K\lambda$. Observing from the optimality condition that $\mathbf{q}^k = \mu\mathbf{A}^*\mathbf{r}^k$ with $\mu = \frac{1}{\tilde{\lambda}}$, the Bregman iteration can be interpreted equivalently as the augmented Lagrangian method for the constrained problem of minimizing $R(\Phi\mathbf{p})$ subject to $\mathbf{y} = \mathbf{A}\mathbf{p}$, i.e.,

$$\mathbf{p}_{k+1} \in \arg\min_{\mathbf{p}} \frac{\mu}{2}\|\mathbf{y} - \mathbf{A}\mathbf{p}\|_{\ell_2}^2 + R(\Phi\mathbf{p}) + \mathbf{r}^k \cdot (\mathbf{y} - \mathbf{A}\mathbf{p})$$

and

$$\mathbf{r}^{k+1} = \mathbf{r}^k + \mathbf{y} - \mathbf{A}\mathbf{p}^{k+1}.$$

While single-pixel cameras are naturally investigated from a compressed sensing point of view, let us comment on some aspects of the problem when viewed as an inverse problem as other image reconstruction tasks in tomography or deblurring. First of all, we can see some common issues in computational techniques, which are needed due to the indirect relation between the image and the measurements and the fact that the matrix $\mathbf{A}$ is related to the discretization of an integral operator. On the other hand, there is a significant difference between the single-pixel setup and other image reconstruction problems in the sense that the adjoint operator is not smoothing, i.e., $\mathbf{A}^*$ has no particular structure. Thus, the typical source condition $\mathbf{A}^*\mathbf{w} \in \Phi^*\partial R(\Phi\mathbf{p})$ for some $\mathbf{w}$ that holds for solutions of the variational problems (cf. [6]) does not imply smoothness of the subgradients as in other inverse problems.

3 Computational Image Reconstruction

In the following, we discuss the effects of different regularization models on the image reconstruction quality. In order to efficiently compute solutions of the arising variational problems at reasonable image resolution, appropriate schemes to handle the convex but nondifferentiable terms are needed. It has become a standard approach in computational imaging of such problems to employ first-order splitting methods based on proximal maps that can be computed exactly, we refer to [13] for an overview. The key idea is to isolate matrix-vector multiplications and local nonlinearities, for which the proximal map can be computed exactly or numerically efficient. A standard example is the ℓ_1-norm, whose ℓ_2-proximal map

$$\operatorname{prox}_{\ell_1}(f) = \arg\min_x \frac{1}{2}\|x - f\|_{\ell_2}^2 + \|x\|_{\ell_1}$$

is given by soft shrinkage. A very popular approach in current computational imaging are first-order primal-dual methods (cf. [18, 61]) for computing minimizers of problems of the form

$$\min_{\mathbf{p}} F(\mathbf{p}) + G(\mathbf{L}\mathbf{p}),$$

with convex functionals F and G and a linear operator $\mathbf{L}$. The primal-dual approach reformulates the minimization as a saddle point problem

$$\min_{\mathbf{p}} \max_{\mathbf{q}} F(\mathbf{p}) + \langle \mathbf{L}\mathbf{p}, \mathbf{q}\rangle - G^*(\mathbf{q}),$$

with G^* being the convex conjugate of G. Then one iterates primal minimization with respect to $\mathbf{p}$ and dual maximization with respect to $\mathbf{q}$ with additional damping and possible extrapolation steps. Here, we use a toolbox by Dirks [24] for rapid prototyping with such methods to implement the models of the previous section.

We will use synthetic data to investigate the regularization methods described above. With this, we can fully explore and understand the effects of named methods for different types of images. Here we consider the multiple regularization operators, namely Tikhonov, ℓ_1- regularization, total variation (TV), and total generalized variation (TGV) [11]. Moreover, we compare the approach with a simple nonnegative least squares approach that only enforces nonnegativity of the image. Each dataset is composed of a ground truth structural image and a Gaussian kernel beam that is used as an illumination source. In this work, we will explore two different ground truth images, that are illuminated by the shown beam. Structural images, beam, and illuminated structures are shown in Fig. 2.

An overview of basic reconstructions in the single-detector framework using random binary masks is shown in Fig. 3. Here, we use two phantoms and two different sampling ratios. We display reconstructions with regularization parameters optimized for SSIM and observe the impact of the regularization for both choices of the sampling rate. In particular, we see that for total variation type regularizations there is a

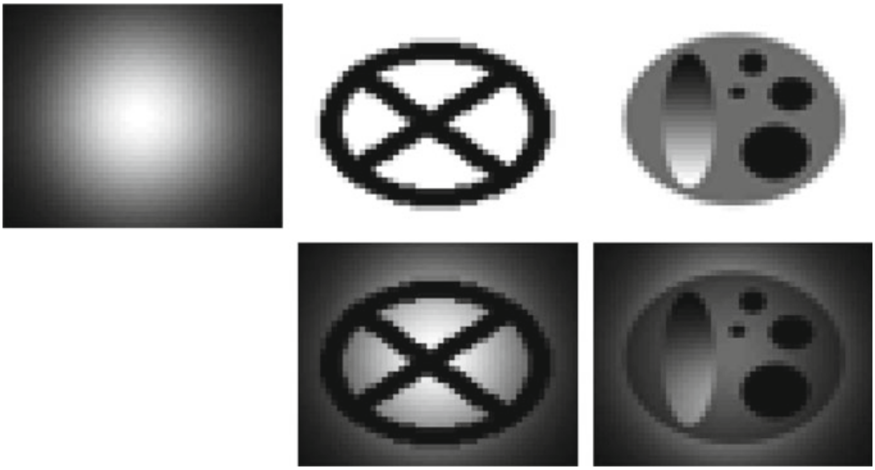


Fig. 2 Grayscale images of synthetic data, synthetic illumination beam with values between 0 (black) and 1 (white); top left: Gaussian beam; top center: x phantom ground truth; bottom center: x phantom with applied beam; top right: gradient phantom ground truth; bottom right: gradient phantom with applied beam

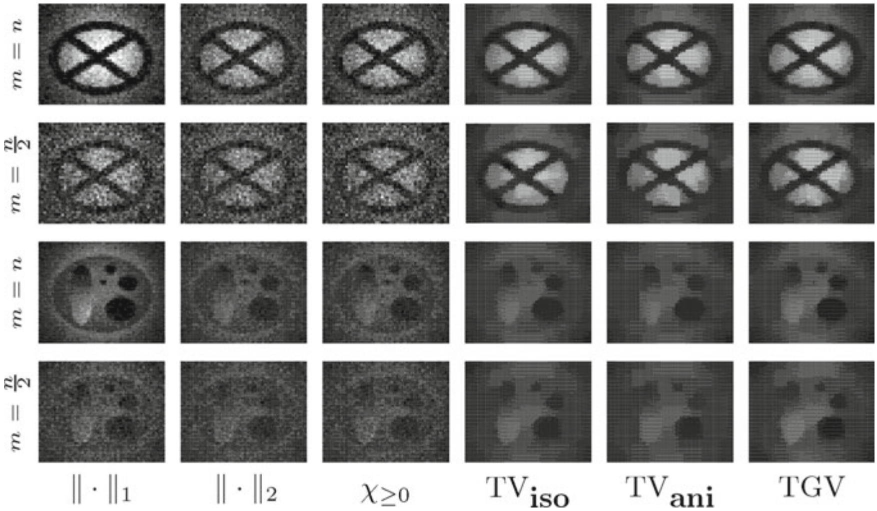


Fig. 3 Reconstruction the x and gradient phantom for multiple sampling rates (rows) and reconstruction frameworks (columns)

strong improvement in visual image quality and that there is hardly any loss when going from full sampling to undersampling.

4 Challenges in Practical THz Single-Pixel Imaging

If compared to the visible region of the electromagnetic spectrum (VIS) the THz region is quite demanding from a physical point of view. Due to the large wavelength that is 2–3 orders of magnitude larger than for VIS radiation, the THz region is plagued by coherence effects and diffraction. This has to be mitigated already on the physical level by using specifically designed optical elements and techniques. As a rule of thumb, techniques and methods from the VIS region can be used but they need to be adapted. Also, lenses and mirrors are used but they are made of different materials or they tend to be bigger. For this reason, the THz region is also sometimes called the quasi-optical or Gaussian region. It is called the Gaussian region because beams in the THz region almost always are of Gaussian shape. This, in turn, means that in the THz region the illumination conditions are non-homogeneous with an exponential decrease of illuminating signal strength towards the edges of the field of view. Due to the large coherence length in this region of the electromagnetic spectrum, one can not simply make the illuminating beam larger and use only the center portion for imaging. This will cause interference effects that appear as dominating artifacts in images (see Fig. 4), which limit image quality and spatial resolution. So, without calibration one either has to live with a limited field of view or one has to accept image artifacts and limited spatial resolution. Therefore, calibration for nonhomogeneous illumination is essential for a practical THz single-pixel imaging system. We will discuss this issue and possible solutions in the next section. Note that for a practical imaging system the calibration of the illuminating beam is very important and also leads to challenges related to the used masks. This is also exemplified in Fig. 5, which shows a 0.35 THz single-pixel camera measurement of a nonmetallic Siemens Star test target reconstructed using a convolution approach with a nonnegative least squares approach, i.e., we use convolutional masks leading to 2D circulant matrices.

In a modulated illumination setting of a THz single-pixel camera, the quality and fidelity of the masks/illumination patterns determine the achievable spatial resolu-

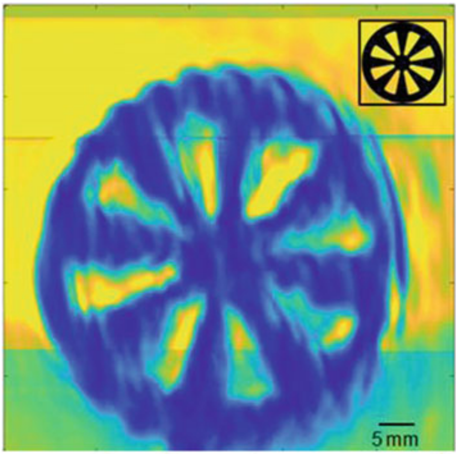


Fig. 4 Mechanically scanned 0.35 THz image of a metal Siemensstar test target with a diameter of 50 mm. The image acquisition took several hours (12 h) depending on the number of steps but still scanning artifacts are very prominent. The inset shows a photo of the metal Siemens Star target

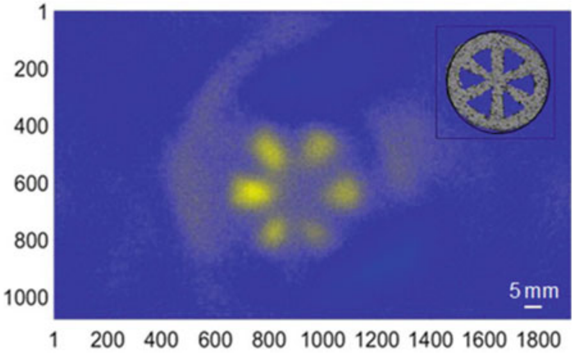


Fig. 5 Megapixel 0.35 THz image using a convolution approach (i.e., a binary circulant matrix). Labels show the pixel numbers in both directions, about 10 pixel correspond to 5 mm. The spatial resolution in the image is still limited by suboptimal illumination patterns but the image shows that the resolution is reasonable using the convolution approach, while being very fast (more than one FPS can be achieved). The inset shows again a photo of the imaged object

tion, the signal-to-noise ratio, the achievable undersampling ratio, and even more. The physical process of implementing the masks is, therefore, very important and potentially introduces deviations into the masks already on the physical level. These potential deviations can be simple blurring effects, the introduction of an offset or the reduction in the so-called modulation depth. As introduced, for binary masks the modulation depth is essentially the difference between ones and zeros. So, the ideal value is of course, 1 or 100% but in a practical system the modulation depth can be several 10% below the ideal value. Depending on the chosen reconstruction approach, this will severely influence the image fidelity of reconstructed images (see Fig. 5 for an example). The example in Fig. 5 shows what happens when the modulation depth is only 40% and the masks are not optimized in an undersampling modality. Due to the fact that the spatial resolution and the signal-to-noise ratio in the image are severely limited the image appears blurry and noisy.

All the aforementioned issues have to be considered on the software level in order to harness the potential power of a THz single-pixel camera and, therefore, often robustness is an important consideration when choosing the reconstruction method. As mentioned, the reconstruction approach used in this example was based on the idea of convolutional mask, which offers strong simplifications in the design of masks and the memory consumptions, moreover the reconstruction algorithms can be made more efficient. Hence, such an approach has a lot of potential for THz single-pixel cameras, but there is still a lot of effort necessary in order to optimize the imaging process.

5 Calibration Problems

In the previous section, we have seen the difficulties to calibrate the illumination beam directly, hence it seems necessary to perform a self-calibration approach during the reconstruction. We hence look for a multiplicative decomposition of the image $\mathbf{p}$ into a smooth light field $\mathbf{d}$ and a normalized target structure $\mathbf{x}$, i.e.,

$$\mathbf{p} = \mathbf{d} \odot \mathbf{x},$$

where $\odot$ denotes pointwise multiplication. A simple approach would be to first reconstruct the image and then use standard decomposition ideas. However, this approach seems suboptimal for undersampled measurements, since better a-priori knowledge on light and the target is available than for their composition.

5.1 Self-calibration and Bilinear Inverse Problems

A very recent trend in compressed sensing is to include the calibration task into the recovery step as well. This approach has been termed also as *self-calibration*. In the case of unknown illumination, this yields a *bilinear inverse problem*. In particular, for sparsity promoting regularizers, this falls in the category of biconvex compressed sensing [47]. In this work, the lifting approach has been adopted to transform the task to an convex formulation, see here also the seminal work on Phaselift [20]. Unfortunately, this approach does scale for imaging problems.

Here, we are confronted with the generic problem:

$$\min_{\mathbf{x}\in[0,1]^n,\mathbf{d}\geq 0} \|\mathbf{y} - \mathbf{A}(\mathbf{d} \odot \mathbf{x})\|_{\ell_2}^2 + \lambda \cdot r(\mathbf{x}, \mathbf{d})$$

with an appropriate regularization functional r for both structures, the illumination $\mathbf{d}$ and the target $\mathbf{x}$. This problem is a particular case of a *bilinear inverse problem* and linked to *compressive blind deconvolution*. Indeed, let us formalize this by using the 2D fast Fourier transform (FFT):

$$\mathbf{d} \odot \mathbf{x} = \frac{1}{\sqrt{n}}\mathbf{F}^{-1}[(\mathbf{F}\mathbf{d}) * (\mathbf{F}\mathbf{x})] = \frac{1}{\sqrt{n}}\mathbf{F}^{-1}[\mathbf{l} * \mathbf{r}]$$

where $\mathbf{F}\colon \mathbb{R}^n \to \mathbb{C}^n$ defines the Fourier transform operator and $\mathbf{l}, \mathbf{r} \in \mathbb{C}^n$. and therefore the observation $\mathbf{y} = \frac{1}{\sqrt{n}}\mathbf{A}\mathbf{F}^{-1}(\mathbf{l} * \mathbf{r})$ is a compressed circular 2D convolution of $\mathbf{l}$ and $\mathbf{r}$. Obviously, without further assumptions this type of inverse problem cannot be solved uniquely.

Let us discuss the case $\mathbf{A} = \mathrm{Id}$ that usually corresponds to an image scanning approach (and not the single-pixel setup). Here, the measurement image is $\mathbf{p}$ and the goal is to factorize it into light $\mathbf{d}$ and a normalized target $\mathbf{x}$ using the program:

$$\min_{\mathbf{x}\in[0,1]^n,\mathbf{d}\geq 0} \|\mathbf{p} - \mathbf{d} \odot \mathbf{x}\|_{\ell_2}^2 + \lambda \cdot r(\mathbf{x}, \mathbf{d}).$$

Note that this formulation is a non-convex problem since it is not jointly convex in $(\mathbf{d}, \mathbf{x})$. To make this problem more well-posed from a compressed viewpoint, further assumptions on $\mathbf{d}$ and $\mathbf{x}$ are necessary. In the case of random subspace assumptions for either $\mathbf{d}$ or $\mathbf{x}$ and some additional incoherence conditions, recovery guarantees for the convexified (lifted) formulation of blind deconvolution have been established in [4]. This framework even extends to the more involved problem of blindly demixing convolutional mixtures, called as also blind demixing and deconvolution, and recently almost-optimal sampling rates could be established here under similar assumptions [38]. However, the analysis of the original non-convex problem (without lifting) is often difficult, requires a priori assumptions and sufficient randomization and the performance of iterative decent algorithms often depends on a good initialization. First results in this direction appeared here for example in [45].

A further way to convexify a related problem is based on a formulation for strictly positive gains

$$\min_{\mathbf{x}\in[0,1]^n, \mathbf{g}>\epsilon} \|\mathbf{p} - \mathbf{g} \odot \mathbf{A}\mathbf{x}\|_{\ell_2}^2 + \lambda \cdot r(\mathbf{x}, \mathbf{g}).$$

In this problem, the image $\mathbf{x}$ is unknown and each measurement has an unknown strictly positive gain $\mathbf{g} > \epsilon$. This problem occurs exactly in our setting when $\mathbf{A} = \mathrm{Id}$ and then $\mathbf{g} = \mathbf{d}$. Thus, under the additional prerequiste of $\mathbf{d} \geq \epsilon > 0$ one could write a constraint $\mathbf{p} = \mathbf{d} \odot \mathbf{x}$ also as $\mathbf{d}^{-1} \odot \mathbf{p} = \mathbf{x}$ which is jointly convex in $(\mathbf{d}^{-1}, \mathbf{x})$ [31, 48]. This approach is also interesting if $\mathbf{x} = \Psi\boldsymbol{\alpha}$ itself is a compressed representation. Unfortunately, this approach does not apply to the single- pixel imaging setup due to matrix $\mathbf{A}$, which cannot be interchanged with the division by $\mathbf{d}$.

5.2 Single-Pixel and Retinex Denoising

A related classical problem in imaging is the so-called Retinex approach. The corresponding problem was first investigated by Land (cf. [43, 46]) in the context of human visual studies and provide a first entry in the developing Retinex theory. The human eye is able to compensate lack of illumination, hence it is able to filter for structural information in the field of view. From a more general perspective, this translates to the question of how to separate an image into a structural and an illumination part. Here, we focus on the approach of Kimmel et al. [39] and further developments of this approach. By defining $\mathbf{s} := \log(\mathbf{p})$, $\mathbf{r} := \log(\mathbf{x})$ and $\mathbf{l} := \log(\mathbf{d})$ we move the image into the logarithmic domain. Again in the case $\mathbf{A} = \mathrm{Id}$ this allows for the usage of a convex variational model as proposed in [39]. The basic assumptions are as follows:

- *spatial smoothness of illumination*,
- $\mathbf{l}$ *is greater than* $\mathbf{s}$: Since $\mathbf{x} \in [0, 1]^n$ and $\mathbf{d} \geq \mathbf{x} \odot \mathbf{d}$ it is $\mathbf{l} \geq \mathbf{s}$ due to the monotone nature of the logarithmic map,
- *non trivial illumination*: resemblance to the original image, hence $l - s$ small,
- *soft smoothing on structure reconstruction* and
- *smooth boundary condition*.

These result in the variation approach proposed by Kimmel et al.:

$$\min_{\mathbf{l}\geq\mathbf{s}} \frac{1}{2}\|\mathbf{l}-\mathbf{s}\|_{\ell_2}^2 + \frac{\alpha}{2}\|\nabla(\mathbf{l}-\mathbf{s})\|_{\ell_2}^2 + \frac{\beta}{2}\|\nabla\mathbf{l}\|_{\ell_2}^2. \tag{6}$$

However, this applies a smoothing on l and $l - s$, which is basically r. Then this boils down to the data fitting term and two smoothing regularization operators that are balanced using the respective parameters α and β. In a later paper by Ng and Wang [51], the usage of TV regularization for the structural image was introduced. The proposed model is

$$\min_{\mathbf{l}\geq\mathbf{s},\mathbf{r}\leq 0} \frac{1}{2}\|\mathbf{l}-\mathbf{s}+\mathbf{r}\|_{\ell_2}^2 + \alpha\|\nabla\mathbf{r}\|_{\ell_1} + \frac{\beta}{2}\|\nabla\mathbf{l}\|_{\ell_2}^2. \tag{7}$$

This equation makes sense from a Retinex point of view. One has given an image $\mathbf{p}$ or $\log(\mathbf{p}) = \mathbf{s}$, respectively, and wants to separate reflection $\log(\mathbf{x}) = \mathbf{r}$ and $\log(\mathbf{d}) = \mathbf{l}$.

A comparison of both Retinex models are shown in Fig. 7, with an interesting parameter dependent behavior. The regularization and weighting parameters α and β have been determined using a parameter test, shown in Fig. 6. We see that choosing parameters optimizing the SSIM measure for the target, respectively, its logarithm $\mathbf{r}$ ($\alpha = 10^0$, $\beta = 10^2$ in the Kimmel model, respectively, $\alpha = 10^1$, $\beta = 10^2$ for the Ng and Wang model) yields a strange result with respect to the illumination, which still has quite some of the target structure included, in addition to the full beam. This can be balanced by different parameter choices ($\alpha = 10^0$, $\beta = 10^3$ in the Kimmel model, respectively, $\alpha = 10^1$, $\beta = 10^3$ for the Ng and Wang model) displayed in the lower line respectively, which eliminates most structure from the reconstructed illumination, but also leaves some beam effects in the target.

However in the framework of single-pixel camera approaches, one does not have the fully recovered image at hand. Instead we only have measured data, hence the data fidelity term in (7) is replaced by a reconstruction based data fidelity term based on the forward operator A and measured data y:

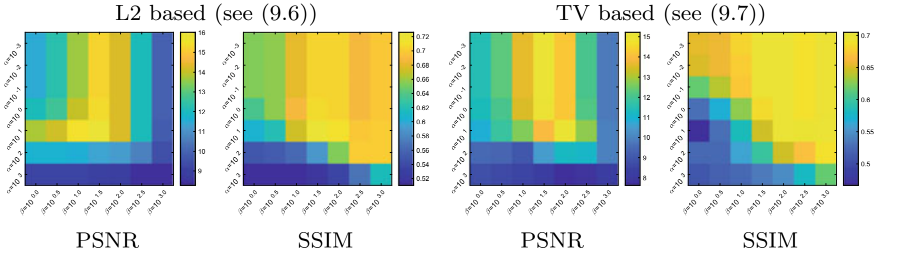


Fig. 6 Overview of PSNR value and SSIM index for systematic parameter test for the synthetic x phantom dataset. Map of regularization parameters in respect to given pair of regularization parameters α and β: α in Y axes with values 10^{-3} (top), 10^{-2}, ..., 10^2, 10^3 (bottom) and β in X axis with values 10^0 (left), $10^{0.5}$, ..., $10^{2.5}$, 10^3 (right)

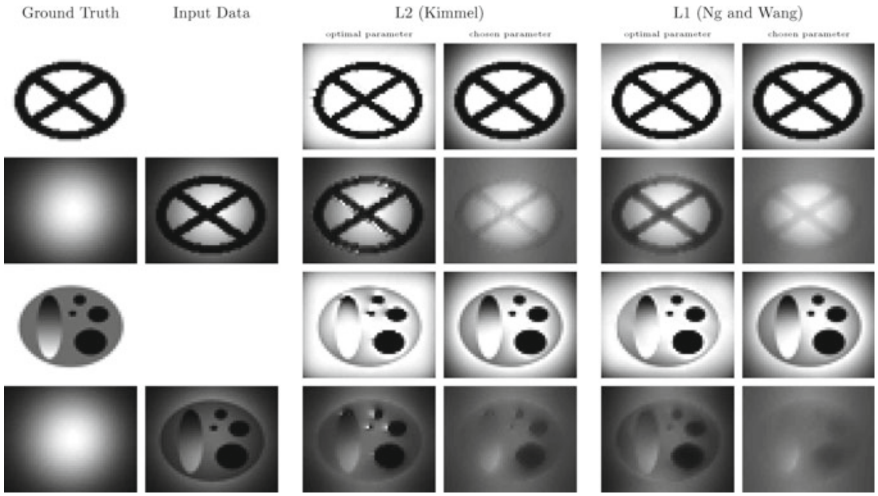


Fig. 7 Comparison of the L2-based Retinex model by Kimmel and the total variation based Retinex model by Ng and Wang for both datasets

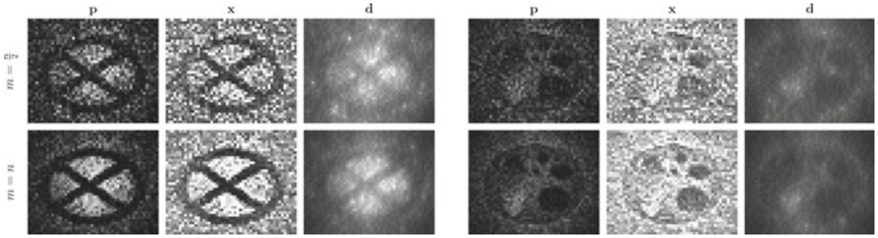


Fig. 8 Reconstruction results for an alternating reconstruction and Retinex scheme

$$\min_{\mathbf{l} \geq \mathbf{s}, \mathbf{r} \leq 0} \|\mathbf{y} - \mathbf{A}(e^{\mathbf{l}+\mathbf{r}})\|_{\ell_2}^2 + \alpha \|\nabla \mathbf{r}\|_{\ell_1} + \frac{\beta}{2} \|\nabla \mathbf{l}\|_{\ell_2}^2.$$

A simple approach is to apply a two-step idea: In the first step, one can compute standard reconstruction of $\mathbf{p}$, e.g., by (4) and subsequently apply the Retinex model with $\mathbf{s} = \log \mathbf{p}$. Since in this case the first reconstruction step does not use prior knowledge about the structure of illumination, it is to be expected that the results are worse compared to a joint reconstruction when the number of measurements decreases.

In order to construct a computational approach for the calibration problem, we still formulate the problem in the logarithmic variables

$$\min_{\mathbf{r} \leq 0, \mathbf{l} \in \mathbb{R}^n} \frac{1}{2} \|\mathbf{y} - \mathbf{A} e^{\mathbf{r}+\mathbf{l}}\|_{\ell_2}^2 + \lambda \cdot \tilde{r}(\mathbf{r}, \mathbf{l})$$

and derive a forward–backward splitting algorithm in these variables: The forward step is simply given by

$$\mathbf{r}^{k+1/2} = \mathbf{r}^k + \tau e^{\mathbf{r}^k+\mathbf{l}^k} \odot \mathbf{A}^T(\mathbf{y} - \mathbf{A}e^{\mathbf{r}^k+\mathbf{l}^k})$$
$$\mathbf{l}^{k+1/2} = \mathbf{l}^k + \tau e^{\mathbf{r}^k+\mathbf{l}^k} \odot \mathbf{A}^T(\mathbf{y} - \mathbf{A}e^{\mathbf{r}^k+\mathbf{l}^k})$$

and the backward step then computes $\mathbf{r}^{k+1}$ as a minimizer of

$$\frac{1}{2\tau}\|\mathbf{r} - \mathbf{r}^{k+1/2}\|_{\ell_2}^2 + \lambda_r\|\nabla r\|_{\ell_1}$$

and $\mathbf{l}^{k+1}$ as a minimizer of

$$\frac{1}{2\tau}\|\mathbf{l} - \mathbf{l}^{k+1/2}\|_{\ell_2}^2 + \lambda_l\|\nabla l\|_{\ell_2}^2.$$

Note that by adding the two equations in the forward step we can directly formulate it as an update for $\mathbf{s}$ as

$$\mathbf{s}^{k+1/2} = \mathbf{s}^k + \tau e^{\mathbf{s}^k} \odot \mathbf{A}^T(\mathbf{y} - \mathbf{A}e^{\mathbf{s}^k}).$$

This induces a simple idea for the two-step approach: we can iterate $\mathbf{s}$ with the above scheme and directly apply the above Retinex model to data $\mathbf{s}^{k+1/2}$. From the resulting minimizers $\mathbf{r}^{k+1}$ and $\mathbf{l}^{k+1}$, we can compute $\mathbf{s}^{k+1} = \mathbf{r}^{k+1} + \mathbf{l}^{k+1}$. The resulting reconstructions are shown in Fig. 8, which are clearly suboptimal since too much of the random structure from the matrix $\mathbf{A}$ is propagated into $\mathbf{s}^{k+1/2}$, which is not reduced enough in the Retinex step. The results of the forward–backward splitting approach are shown in Fig. 9, which are clearly improving the separation of illumination beam and target and yield robustness with respect to undersampling, although still not providing perfect smoothing of the images. This might be expected from improved regularization models for the decomposition to be investigated in the future.

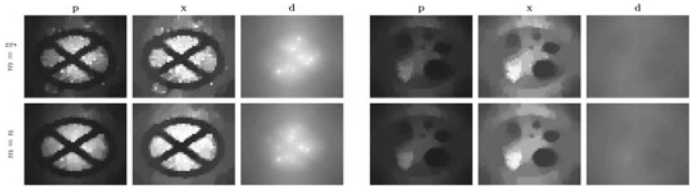


Fig. 9 Reconstruction results for Forward–Backward Splitting including the L1-based model proposed by Ng and Wang

6 Phase Retrieval in Single-Pixel Cameras

In the following, we discuss another aspect of reconstruction in single-pixel cameras, namely, phase reconstruction problems caused by diffraction effects. This problem is a particular instance of the difficult phase retrieval problem. Compared to blind deconvolution, as a (non-convex) bilinear inverse problem, phase retrieval is the corresponding quadratic (non-convex) case and therefore most analytical results here are based on lifting the problem to a convex formulation [20]. For some overview, further references and due to limited space we refer here exemplary to the overview article [36]. Interestingly also, first works already appeared where phase retrieval and blind deconvolution are combined using lifting [1]. However, already for imaging problems at moderate resolution these approaches usually not scale and require a priori random subspace assumptions which are difficult to fulfill in practice. In the following we will discuss how to setup the phase reconstruction problem in the diffraction case.

The propagation of light waves can—after some approximation—be represented by the Fresnel diffraction integral, we refer to [50] and references cited therein for a detailed treatment of the discretization problem. In the single-pixel setup (compare Fig. 1), we consider that the diffraction has to be taken into account in two ways, namely, between the object and mask plane and the mask and detector plane. This means that the measurement matrix is further changed by the introduction of the masks. In addition to this, only the magnitude of the combined complex signals is obtained. Let $\mathbf{D}_{om}$ be a matrix discretizing the diffraction integral from object to mask plane and $\mathbf{D}_{md}$ be a matrix discretizing the diffraction from mask to detector plane. Then the complex signal arriving at the detector plane is

$$\mathbf{z}_i = \mathbf{1}^T \mathbf{D}_{md} \operatorname{diag}(\mathbf{A}_i) \mathbf{D}_{om} \mathbf{p},$$

where $\mathbf{p}$ is the complex image, $\mathbf{A}_i$ denotes the ith row of $\mathbf{A}$, and $\mathbf{1}$ a vector filled with all entries equal to one. Note that in the absence of diffraction, i.e., $\mathbf{D}_{md} = \mathbf{D}_{om} = \mathbf{Id}$, this reduces to the standard single- pixel camera approach discussed above. The measured intensity is then the absolute value of $\mathbf{z}_i$, or rather its square, i.e.,

$$\mathbf{y}_i = |\mathbf{z}_i|^2 = |\mathbf{1}^T \mathbf{D}_{md} \operatorname{diag}(\mathbf{A}_i) \mathbf{D}_{om} \mathbf{p}|^2.$$

Defining a matrix $\mathbf{B}$ with rows

$$\mathbf{B}_i = \mathbf{1}^T \mathbf{D}_{md} \operatorname{diag}(\mathbf{A}_i) \mathbf{D}_{om},$$

the phase reconstruction problem can be written in compact notation as

$$\mathbf{y} = |\mathbf{B}\mathbf{p}|^2. \tag{8}$$

It is apparent that (8) is a nonlinear problem compared to the linear phase- and diffractionless problem, where both the matrix $\mathbf{A}$ and the image $\mathbf{p}$ can be modeled to be real nonnegative (hence the absolute value does not lead to a loss of information). For this reason, the iterative solution of (8), respectively, least-squares versions thereof is a problem of central importance, which we will discuss in the next section.

6.1 Algorithms for Phase Retrieval

Phase retrieval is a classical problem in signal processing, which has been revived by compressed sensing approaches in the past decade, cf. [7, 29, 35, 49] for an overview of classical and recent methods. A famous early algorithm is the Gerchberg–Saxton (GS) algorithm (cf. [32]). However, a version of the GS algorithm has been developed by Fienup (cf. [29]) that works with amplitude measurements only and is using a multiplicative update scheme of the form

$$\mathbf{y}_{k+1} = \frac{|\mathbf{y}|}{|\mathbf{B}\mathbf{p}_k|}\,\mathbf{B}\mathbf{p}_k, \quad \mathbf{p}_{k+1} = \mathbf{B}^{+}\mathbf{y}_{k+1},$$

where $\mathbf{B}^{+}$ defines the commonly used pseudoinverse of $\mathbf{B}$ (cf. [27]). In the case of additional constraints on $\mathbf{p}$, those are applied to modify $\mathbf{y}_{k+1}$ in an additional projection step. Note that the original Gerchberg–Saxton algorithm is formulated for Fourier measurements only, where $\mathbf{B}$ is invertible by its adjoint. The algorithm can be used in particular to compute real images, where

$$\mathbf{p}_{k+1} = \text{Re}\ (\mathbf{B}^{+}\mathbf{y}_{k+1}).$$

Another approach that can be found at several instances in literature (cf. [10, 34, 55]) is based on the application of (variants of) Gauss–Newton methods to the least squares problem of minimizing

$$L(\mathbf{p}) = \|\mathbf{y} - |\mathbf{B}\mathbf{p}|^2\|^2.$$

This amounts to linearizing the residuals around the last iterate and to obtain $\mathbf{p}_{k+1}$ as the minimizer of

$$\sum_{i=1}^{m} |\mathbf{y}_i - |\mathbf{B}_i \cdot \mathbf{p}_k|^2 + 2(\text{Re}(\mathbf{B}_{\mathbf{i}})\text{Re}(\mathbf{B}_{\mathbf{i}}) \cdot \mathbf{p}_k + \text{Im}(\mathbf{B}_{\mathbf{i}})\text{Im}(\mathbf{B}_i) \cdot \mathbf{p}_k) \cdot (\mathbf{p} - \mathbf{p}_k)|^2.$$

Several variants are used to stabilize the Gauss–Newton iteration and to account for the ill-conditioning of the Jacobian matrix

$$\mathbf{J}_k = 2\left((\text{Re}(\mathbf{B}_{\mathbf{i}})\text{Re}(\mathbf{B}_{\mathbf{i}}) \cdot \mathbf{p}_k + \text{Im}(\mathbf{B}_{\mathbf{i}})\text{Im}(\mathbf{B}_{\mathbf{i}}) \cdot \mathbf{p}_k)\right)_{i=1,\ldots,m} \cdot$$

A popular one, which we will also use in our numerical tests, is the Levenberg–Marquardt method, which computes $\mathbf{p}_{k+1}$ as the minimizer of

$$\|\mathbf{y} - |\mathbf{B}\mathbf{p}_k|^2 + \mathbf{J}_k(\mathbf{p} - \mathbf{p}_k)\|^2 + \alpha_k \|\mathbf{p} - \mathbf{p}_k\|^2,$$

with α_k a decreasing sequence of positive parameters.

A recently popular approach to solve the non-convex least-squares problem is the Wirtinger flow (cf. [16]). Its main ingredient is just gradient descent on the least squares functional L, i.e.,

$$\begin{aligned}\mathbf{p}_{k+1} &= \mathbf{p}_k - \frac{\mu_{k+1}}{2m\|\mathbf{p}_0\|^2}\nabla L(\mathbf{p}_k)\\ &= \mathbf{p}_k + \frac{\mu_{k+1}}{m\|\mathbf{p}_0\|^2}\sum_{i=1}^{m}(\mathbf{y}_i - |\mathbf{B}_i \cdot \mathbf{p}_k|^2)(\mathrm{Re}(\mathbf{B}_i\mathbf{B}_i^T) + \mathrm{Im}(\mathbf{B}_i\mathbf{B}_i^T)).\end{aligned}$$

For the choice of the step size, an increasing strategy like $\mu_k \sim 1 - e^{-k/k_0}$ is proposed. Obviously, the gradient descent as well as the Levenberg–Marquardt method above rely on appropriate initializations in order to converge to a global minimum. For this sake a spectral initialization has been proposed, which chooses p_0 as the first eigenvector of the positive semidefinite matrix

$$\mathbf{M}_0 = \sum_{i=1}^{m} \mathbf{y}_i(\mathrm{Re}(\mathbf{B}_i\mathbf{B}_i^T) + \mathrm{Im}(\mathbf{B}_i\mathbf{B}_i^T),$$

corresponding to the data sensitivity in the least squares functional. For practical purposes, the first eigenvector can be approximated well with the power method.

A very recent approach is the truncated amplitude flow (cf. [58]), whose building block is gradient descent on the alternative least squares functional

$$\tilde{L}(\mathbf{p}) = \sum_{i=1}^{m} |\sqrt{\mathbf{y}_i} - |\mathbf{B}_i^T\mathbf{p}||^2,$$

which is however not differentiable for $\mathbf{B}_i^T\mathbf{p} = 0$. In the case the derivative exists we find

$$\tilde{L}(\mathbf{p}) = -\sum_{i=1}^{m}(\sqrt{\mathbf{y}_i} - |\mathbf{B}_i^T\mathbf{p}|)\frac{1}{|\mathbf{B}_i^T\mathbf{p}|}\mathbf{B}_i\mathbf{B}_i^T\mathbf{p}.$$

The truncated amplitude flow now only selects a part of the gradient in order to avoid the use of small $|\mathbf{B}_i^T\mathbf{p}|$ (compared to $\sqrt{\mathbf{y}_i}$), i.e.,

$$\mathbf{p}_{k+1} = \mathbf{p}_k + \mu_k \sum_{i \in I_k}\left(\frac{\sqrt{\mathbf{y}_i}}{|\mathbf{B}_i^T\mathbf{p}_k|} - 1\right)\mathbf{B}_i\mathbf{B}_i^T\mathbf{p}_k,$$

with index set

$$I_k = \left\{ i \in \{1, \dots, m\} \mid \frac{\sqrt{\mathbf{y}_i}}{|\mathbf{B}_i^T \mathbf{p}_k|} \leq 1 + \gamma \right\}$$

with some $\gamma > 0$. For the truncated amplitude flow, another initialization strategy has been proposed in [58], which tries to minimize the angles of $\mathbf{p}_0$ to the measurement vectors $\mathbf{B}_i$ for a subset of indices I_0. This is equivalent to computing the eigenvector for the smallest eigenvalue of the matrix

$$\tilde{\mathbf{M}}_0 = \sum_{i \in I_0} \frac{1}{\|\mathbf{B}_i\|^2} \mathbf{B}_i \mathbf{B}_i^T.$$

Since computing such an eigenvector is a problem of potentially high effort, it is approximated by computing the largest eigenvalue of a matrix built of the remaining rows of $\mathbf{B}$ (cf. [58] for further details).

In order to introduce some regularization into any of the flows above, we can employ a forward–backward splitting approach. For example, we can produce the Wirtinger flow to produce the forward estimate and use an additional backward proximal step

$$\begin{aligned} \mathbf{p}_{k+1/2} &= \mathbf{p}_k - \frac{\mu_{k+1}}{2m\|\mathbf{p}_0\|^2} \nabla L(\mathbf{p}_k) \\ \mathbf{p}_{k+1} &= \operatorname{prox}_{\lambda_{k+1} R}(\mathbf{p}_{k+1/2}) \end{aligned}$$

with the regularization functional R and a parameter λ_{k+1} chosen appropriately in dependence of μ_{k+1}. The proximal map $\mathbf{p} = \operatorname{prox}_{\lambda_{k+1} R}(\mathbf{p}_{k+1/2})$ is given as the unique minimizer of

$$\frac{1}{2}\|\mathbf{p} - \mathbf{p}_{k+1/2}\|^2 + \lambda_{k+1} R(\mathbf{p}).$$

Finally, we mention that in addition to the non-convex minimization approaches there has been a celebrated development toward convexifying the problem, the so-called PhaseLift approach, which can be shown to yield an exact convex relaxation under appropriate conditions (cf. [20]). The key idea is to embed the problem into the space of $n \times n$ matrices and to find $\mathbf{p}\mathbf{p}^T$ as a low rank solution of a semidefinite optimization problem. The squared absolute value is rewritten as

$$|\mathbf{B}_i \mathbf{p}|^2 = \mathbf{B}_i^* \mathbf{P} \mathbf{B}_i$$

with the positive semidefinite rank-one matrix $\mathbf{P} = \mathbf{p}\mathbf{p}^*$. The system of quadratic equations $|\mathbf{B}_i \mathbf{p}|^2 = y_i$ then is rewritten as the problem of finding the semidefinite matrix of lowest rank that solves the linear matrix equations $\mathbf{B}_i^* \mathbf{P} \mathbf{B}_i = y_i$. Under appropriate condition, this problem can be exactly relaxed to minimizing the nuclear norm of $\mathbf{P}$, a convex functional, subject to linear equation and positive semidefiniteness. The complexity of solving the semidefinite problem in dimension n^2 makes this approach prohibitive for applications with large n however, hence we will not use it in our computational study in the next section.

6.2 Results

In the following, we present some results obtained from the algorithms discussed above when applied to our single-pixel setup and compare their performance. We start by using the algorithms from the previous section for reconstructing a real-valued image showing the figure π ($n = 900$) without noise in the measurements. For this sake, we consider different types of undersampling and three different initializations: a random initial value (rand), the spectral initialization proposed originally for the Wirtinger flow (spec) and the orthogonality promoting initialization originally proposed for the truncated amplitude flow (amp). The results for the four different methods are shown in Figs. 10 and 11. The results clearly demonstrate the

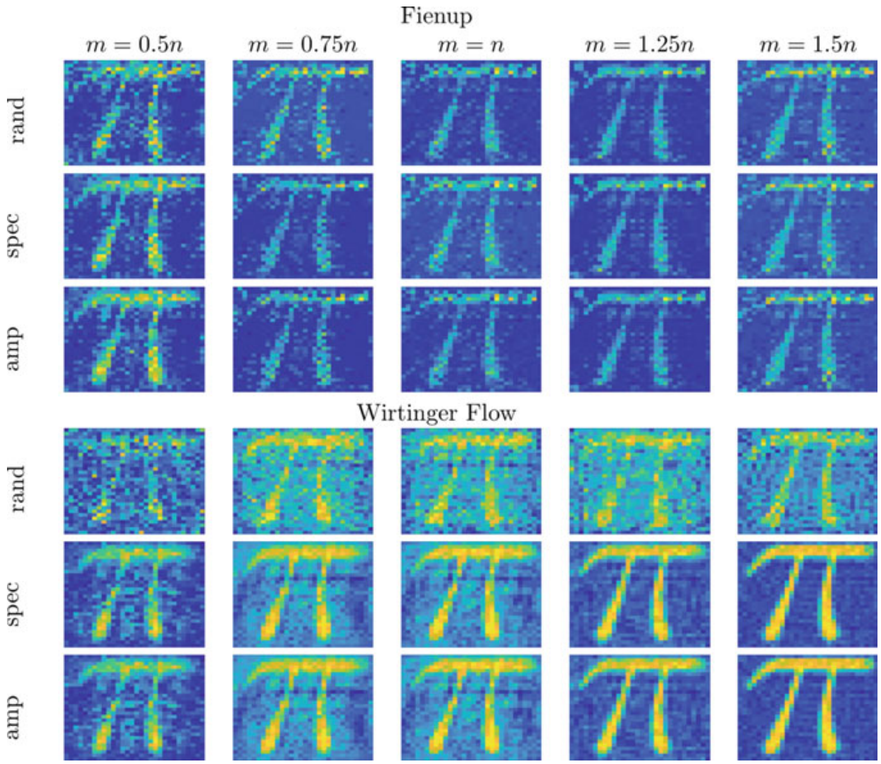


Fig. 10 Reconstructions of π-image with the Fienup-variant of the Gerchberg–Saxton algorithm and the Wirtinger flow (from [50])

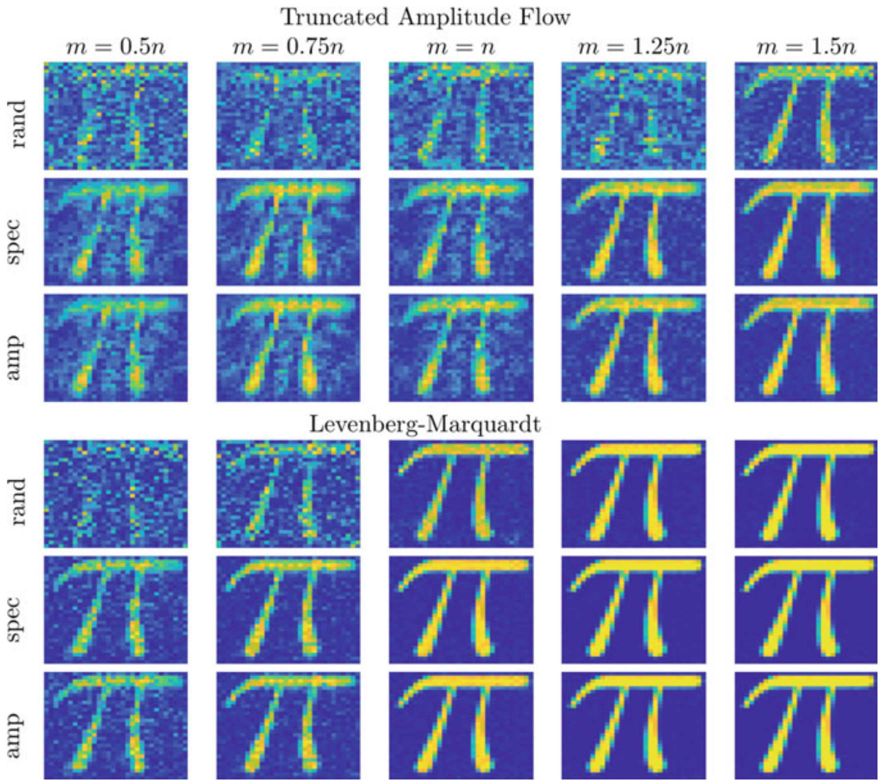


Fig. 11 Reconstructions of π-image with the truncated amplitude flow and the Levenberg–Marquardt method (from [50])

dependence of solutions on the initial value, in particular we see the improvement of the two new initialization strategies with respect to the random one (except in the case of the algorithm by Fienup, which however yields inferior results in all cases). We observe that the Levenberg–Marquardt method performs particularly well for larger sampling rates and yields an almost perfect reconstruction, but also produces suitable reconstructions for stronger undersampling. A more quantitative evaluation is given in Fig. 12, which provides plots of the relative mean-square error versus the sampling rate, surprisingly the Levenberg–Marquardt method outperforms the other schemes for most rates.

The positive effect of regularization and its necessity for very strong undersampling is demonstrated in Fig. 13, where we display the reconstruction results obtained with the forward–backward splitting strategy and the total variation as regularization functional. We see that both approaches yield similar results and in particular allow to proceed to very strong undersampling with good quality results. The effect of noise in the data, here for a fixed sampling ratio $\frac{m}{n} = 0.7$ is demonstrated in Fig. 14

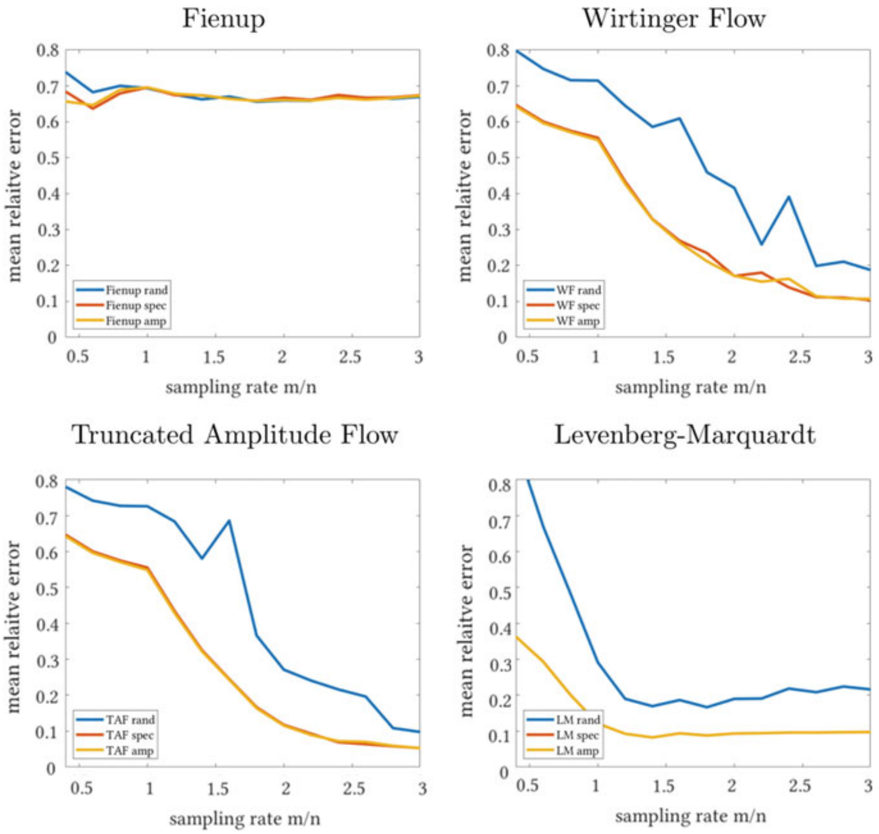


Fig. 12 Plot of relative mean-square error to ground truth versus sampling rate for reconstructions of π-image (from [50])

for different signal-to-noise ratios. Again, not surprisingly, the regularized version of the flows yields stable reconstructions of good quality even for data of lower quality.

Let us finally turn our attention to the reconstruction of the phase in a complex image. The ground truth for the amplitude and phase of the complex signal are shown in Fig. 15. For brevity, we only provide visualizations of reconstructions for the truncated amplitude flow and the Levenberg–Marquardt method in Fig. 16, which clearly indicates that the truncated amplitude flow outperforms the Levenberg–Marquardt method, which is the case for all conducted experiments. Again, a more quantitative evaluation is given in Fig. 17, which provides a plot of the relative mean-square error (made invariant to a global phase that is not identifiable) versus the sampling rate. We see that the Wirtinger flow can obtain the same reconstruction quality for very low sampling rates, but is outperformed by the truncated amplitude flow at higher sampling rates.

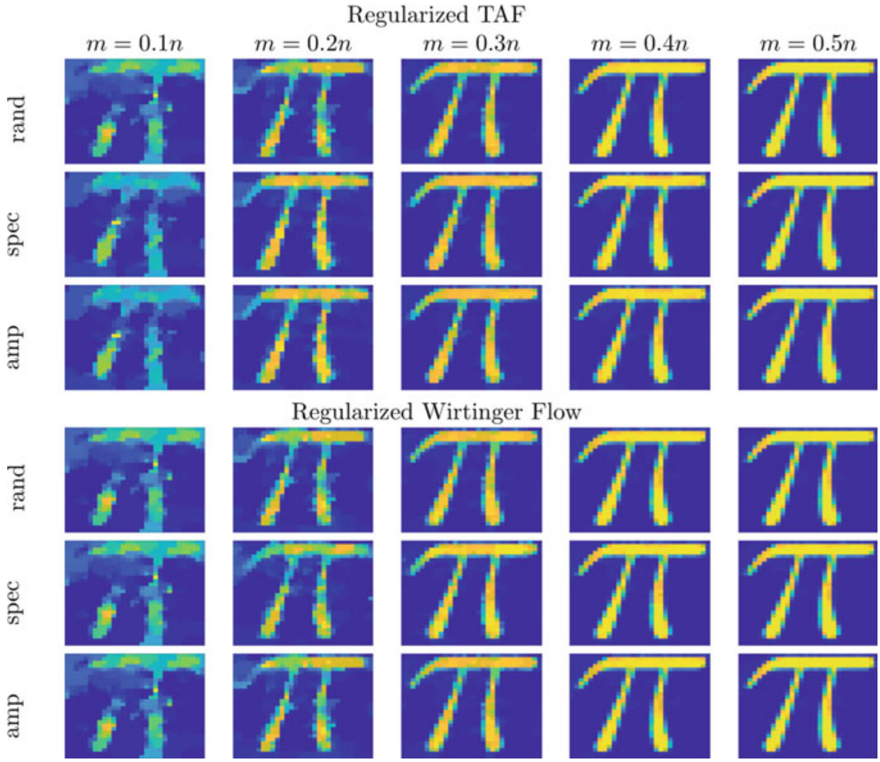


Fig. 13 Reconstructions of π-image with the regularized version of the truncated amplitude flow and the Wirtinger flow (from [50])

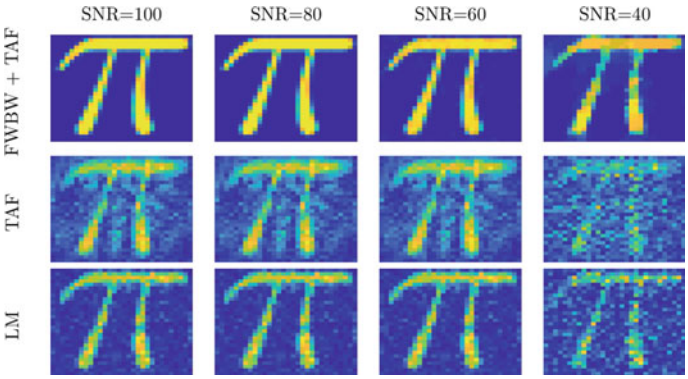


Fig. 14 Reconstructions of π-image from noisy data with different signal-to-noise ratios (SNR) (from [50])

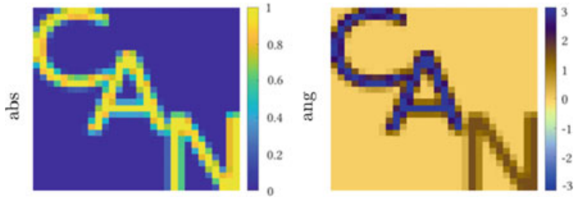


Fig. 15 Ground truth for amplitude (left) and phase (right, from [50])

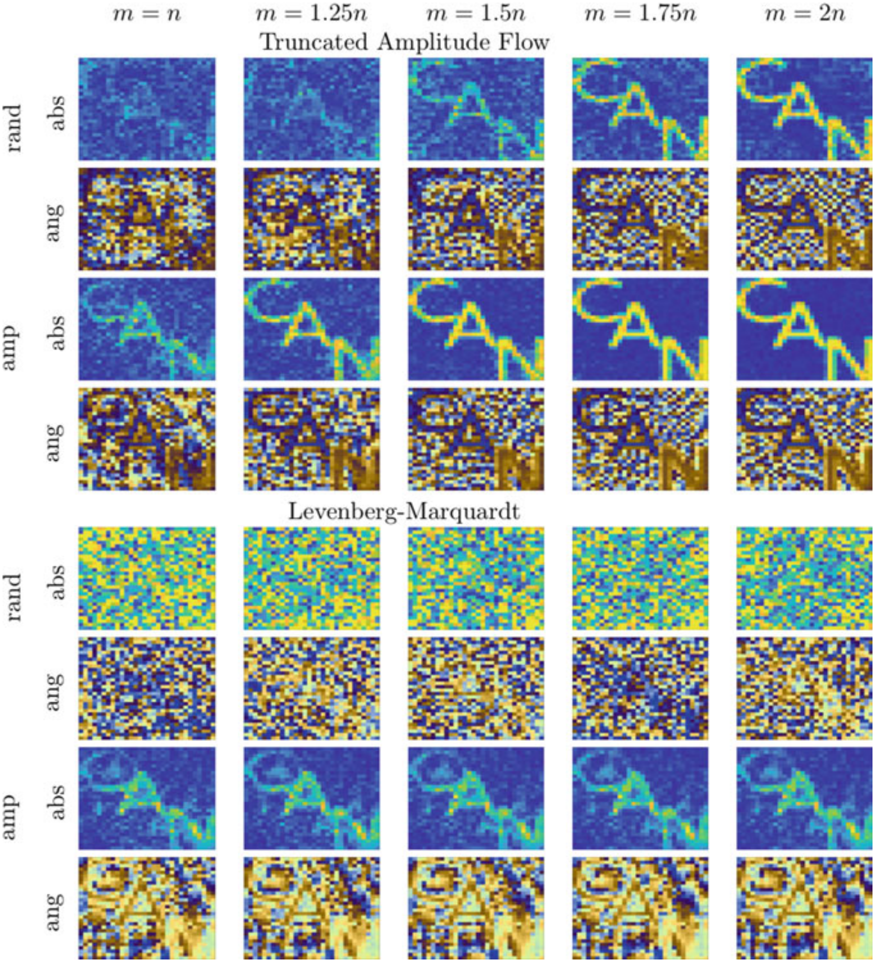


Fig. 16 Reconstructions of amplitude (left) and phase (right, from [50]) with truncated amplitude flow and Levenberg–Marquardt method with different initializations and sampling rates

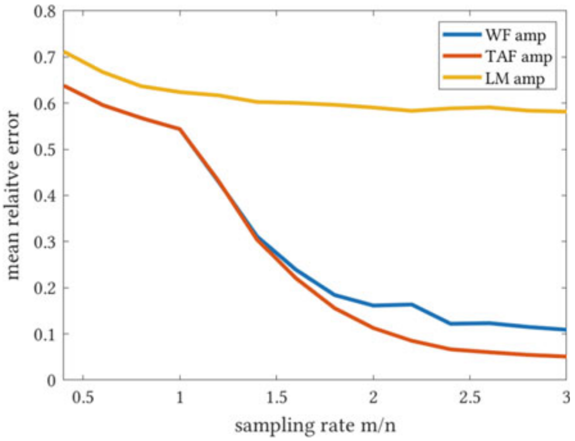


Fig. 17 Relative mean-square error in the reconstructions of the complex image (from [50])

7 Conclusion

We have seen that the compressed sensing approach based on single-pixel imaging has great potential to decrease measurement time and effort in THz imaging, but the application in practice depends on several further challenges to be mastered. First of all appropriate regularization models and numerical algorithms are needed for the image reconstruction problem in order to obtain higher image resolution at reasonable computational times. Moreover, in several situations it is crucial to consider auto-calibration issues in particular related to the fact that the illuminating beam is difficult to be characterized, and in some cases also diffraction becomes relevant, which effectively yields a phase retrieval problem. Both effects change the image reconstruction from a problem with a rather simple and incoherent linear forward model to a nonlinear problem with a more coherent forward model, which raises novel computational and also theoretical issues, since the assumptions of the existing compressed sensing theory are not met.

Besides the above-mentioned issues several further aspects of practical imaging are foreseen to become relevant for THz single- pixel imaging, examples being motion correction for imaging in the field or of moving targets and the reconstruction of multi- or hyperspectral images, which is a natural motivation in the THz regime. In the latter case, it will become a central question how to combine as few masks as possible in different spectral locations.

Acknowledgements This work has been supported by the German research foundation (DFG) through SPP 1798, Compressed Sensing in Information Processing, projects BU 2327/4-1 and JU 2795/3. MB acknowledges further support by ERC via Grant EU FP7 ERC Consolidator Grant 615216 LifeInverse.

References

1. A. Ahmed, A. Aghasi, P. Hand, Blind deconvolutional phase retrieval via convex programming. CoRR (2018)
2. S. Augustin, S. Frohmann, P. Jung, H.-W. Hübers, An optically controllable 0.35 THz single-pixel camera for millimeter resolution imaging. In *2017 42nd International Conference on Infrared, Millimeter, and Terahertz Waves (IRMMW-THz)* (IEEE, 2017), pp. 1–2
3. S. Augustin, J. Hieronymus, P. Jung, H.-W. Hübers, Compressed sensing in a fully non-mechanical 350 ghz imaging setting. J. Infrared Millim. Terahertz Waves **36**(5), 496–512 (2015)
4. A. Ahmed, J. Romberg, B. Recht, Blind deconvolution using convex programming. IEEE Trans. Inf. Theory **60**(3), 1711–1732 (2014)
5. R.G. Baraniuk, Compressive sensing. IEEE Signal Process. Mag. **24**(4), 118–121 (2007)
6. M. Benning, M. Burger, Modern regularization methods for inverse problems. Acta Numer. **27**, 1–111 (2018)
7. H.H. Bauschke, P.L. Combettes, D. Russell Luke. Phase retrieval, error reduction algorithm, and fienup variants: a view from convex optimization. JOSA A, **19**(7), 1334–1345 (2002)
8. M. Burger, H. Dirks, C.-B. Schonlieb, A variational model for joint motion estimation and image reconstruction. SIAM J. Imaging Sci. **11**(1), 94–128 (2018)
9. A.M. Bruckstein, M. Elad, M. Zibulevsky, On the uniqueness of non-negative sparse & redundant representations, in *IEEE International Conference on Acoustics, Speech and Signal Processing* (IEEE, 2008), pp 5145–5148
10. B. Blaschke-Kaltenbacher, H.W. Engl, Regularization methods for nonlinear ill-posed problems with applications to phase reconstruction, in *Inverse Problems in Medical Imaging and Nondestructive Testing* (Springer, 1997), pp. 17–35
11. K. Bredies, K. Kunisch, T. Pock, Total generalized variation. SIAM J. Imaging Sci. **3**(3), 492–526 (2010)
12. M. Burger, S. Osher. A guide to the tv zoo, in *Level Set and PDE Based Reconstruction Methods in Imaging* (Springer, 2013), pp 1–70
13. M. Burger, A. Sawatzky, G. Steidl, First order algorithms in variational image processing, in *Splitting Methods in Communication, Imaging, Science, and Engineering* (Springer, 2016), pp. 345–407
14. V. Caselles, A. Chambolle, M. Novaga, Total variation in imaging, in *Handbook of Mathematical Methods in Imaging*, pp. 1–39 (2014)
15. W.L. Chan, K. Charan, D. Takhar, K.F. Kelly, R.G. Baraniuk, D.M. Mittleman, A single-pixel terahertz imaging system based on compressed sensing. Appl. Phys. Lett. **93**(12), 121105 (2008)
16. E.J. Candes, X. Li, M. Soltanolkotabi, Phase retrieval via wirtinger flow: theory and algorithms. IEEE Trans. Inf. Theory **61**(4), 1985–2007 (2015)
17. E.J. Candes, Y. Plan, A probabilistic and ripless theory of compressed sensing. IEEE Trans. Inf. Theory **57**(11), 7235–7254 (2011)
18. A. Chambolle, T. Pock, A first-order primal-dual algorithm for convex problems with applications to imaging. J. Math. Imaging Vis. **40**(1), 120–145 (2011)
19. E.J. Candes, J. Romberg, T. Tao, Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. IEEE Trans. Inform. Theory **52**(2), 489–509 (2006)
20. E.J. Candes, T. Strohmer, V. Voroninski, Phaselift: exact and stable signal recovery from magnitude measurements via convex programming. Commun. Pure Appl. Math. **66**(8), 1241–1274 (2013)
21. J.-F. Cai, W. Xu, Guarantees of total variation minimization for signal recovery. Inf. Inference 328–353 (2015)
22. D.L. Donoho, et al, Compressed sensing. IEEE Trans. Inf. Theory **52**(4), 1289–1306 (2006)
23. M.F. Duarte, M.A. Davenport, D. Takhar, J.N. Laska, T. Sun, K.F. Kelly, R.G. Baraniuk, Single-pixel imaging via compressive sampling. IEEE Signal Process. Mag. **25**(2), 83–91 (2008)

24. H. Dirks, A flexible primal-dual toolbox. arXiv:1603.05835 [cs, math] (2016)
25. D.L. Donoho, Compressed sensing. IEEE Trans. Inform. Theory **52**(4), 1289–1306 (2006)
26. M.P. Edgar, G.M. Gibson, M.J. Padgett, Principles and prospects for single-pixel imaging. Nat. Photonics 1 (2018)
27. H.W. Engl, M. Hanke, A. Neubauer, *Regularization of Inverse Problems*, vol. 375, 1st edn. Springer Netherlands (2000)
28. Y.C. Eldar, G. Kutyniok, *Compressed Sensing: Theory and Applications*. Cambridge University Press (2012)
29. J.R. Fienup, Phase retrieval algorithms: a comparison. Appl. Opt. **21**(15), 2758–2769 (1982)
30. S. Foucart, H. Rauhut, *A Mathematical Introduction to Compressive Sensing* Appl. Numer. Harmon. Anal. Birkhäuser, Springer, New York (2013)
31. R. Gribonval, G. Chardon, L. Daudet, Blind calibration for compressed sensing by convex optimization, in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (IEEE, 2012), pp. 2713–2716
32. R.W. Gerchberg, A practical algorithm for the determination of phase from image and diffraction plane pictures. Optik **35**, 237–246 (1972)
33. K. Guo, D. Labate, Optimally sparse multidimensional representation using shearlets. SIAM J. Math. Anal. **39**(1), 298–318 (2007)
34. B. Gao, X. Zhiqiang, Phaseless recovery using the Gauss-Newton method. IEEE Trans. Signal Process. **65**(22), 5885–5896 (2017)
35. K. Jaganathan, Y.C. Eldar, B. Hassibi, Phase retrieval: an overview of recent developments. arXiv preprint arXiv:1510.07713 (2015)
36. K. Jaganathan, Y.C. Eldar, B. Hassibi, Phase retrieval: an overview of recent developments, in *Optical Compressive Imaging* ed. by A. Stern (CRC Press, 2016) pp. 263–287
37. P. Jung, R. Kueng, D.G. Mixon, Derandomizing compressed sensing with combinatorial design. CoRR (2018)
38. P. Jung, F. Krahmer, D. Stoeger, B. Demixing, Deconvolution at near-optimal rate. IEEE Trans. Inf. Theory **64**(2), 704–727 (2018)
39. R. Kimmel, M. Elad, D. Shaked, R. Keshet, I. Sobel, A variational framework for retinex. International Journal of Computer Vision **52**(1), 7–23 (2003)
40. R. Kueng, P. Jung, Robust nonnegative sparse recovery and the nullspace property of 0/1 measurements. IEEE Trans. Inf. Theory **64**(2), 689–703 (2017)
41. F. Krahmer, C. Kruschel, M. Sandbichler, Total variation minimization in compressed sensing, in *Compressed Sensing and its Applications* (Springer, 2017), pp. 333–358
42. G. Kutyniok, D. Labate, *Shearlets: Multiscale analysis for multivariate data*. Springer Science & Business Media (2012)
43. E.H. Land, The retinex. Am. Sci. **52**(2), 247–264 (1964)
44. L. Weizhi, W. Li, K. Kpalma, J. Ronsin, Compressed sensing performance of random Bernoulli matrices with high compression ratio. IEEE Signal Process. Lett. **22**(8), 1074–1078 (2015)
45. X. Li, S. Ling, T. Strohmer, K. Wei, Rapid, robust, and reliable blind deconvolution via nonconvex optimization. Appl. Comput. Harmon. Anal. 1–49 (2018)
46. E.H. Land, J.J. McCann, Lightness and retinex theory. J. Opt. Soc. Am. **61**(1), 1 (1971)
47. S. Ling, T. Strohmer, Self-calibration and biconvex compressive sensing. Inverse Probl. **31**(11) (2015)
48. S. Ling, T. Strohmer, Self-calibration and bilinear inverse problems via linear least squares. SIAM J. Imaging Sci. **11**(1), 252–292 (2018)
49. S. Marchesini, A unified evaluation of iterative projection algorithms for phase retrieval. Rev. Sci. Instrum. **78**(1), 011301 (2007)
50. L. Nickel, Phase retrieval in single detector cameras. Master's thesis, WWU Münster (2018)
51. M.K. Ng, W. Wang, A total variation model for retinex. SIAM J. Imaging Sci. **4**(1), 345–365 (2011)
52. D. Needell, R. Ward, Near-optimal compressed sensing guarantees for total variation minimization. IEEE Trans. Image Process. **22**(10), 3941–3949 (2013)

53. D. Needell, R. Ward, Stable image reconstruction using total variation minimization. SIAM J. Imaging Sci. **6**(2), 1035–1058 (2013)
54. C. Poon, On the role of total variation in compressed sensing. SIAM J. Imaging Sci. **8**(1), 682–720 (2015)
55. Y. Shechtman, A. Beck, Y.C. Eldar, Gespar: Efficient phase retrieval of sparse signals. IEEE Trans. Signal Process. **62**(4), 928–938 (2014)
56. M. Slawski, M. Hein, Non-negative least squares for high-dimensional linear models: Consistency and sparse recovery without regularization. Electron. J. Statist. **7**, 3004–3056 (2013)
57. R.J. Stokoe, P.A. Stockton, A. Pezeshki, R.A. Bartels, Theory and applications of structured light single pixel imaging, in *Three-Dimensional and Multidimensional Microscopy: Image Acquisition and Processing XXV*, vol. 10499 (International Society for Optics and Photonics, 2018), p. 104990E
58. G. Wang, G.B. Giannakis, Y.C. Eldar, Solving systems of random quadratic equations via truncated amplitude flow. IEEE Trans. Inf. Theory **64**(2), 773–794 (2018)
59. C.M. Watts, D. Shrekenhamer, J. Montoya, G. Lipworth, J. Hunt, T. Sleasman, S. Krishna, D.R. Smith, W.J. Padilla, Terahertz compressive imaging with metamaterial spatial light modulators. Nat. Photonics **8**(8), 605 (2014)
60. M. Wang, X. Weiyu, A. Tang, A unique "nonnegative" solution to an underdetermined system: from vectors to matrices. IEEE Trans. Inform. Theory **59**(3), 1007–1016 (2011)
61. X. Zhang, M. Burger, S. Osher, A unified primal-dual algorithm framework based on Bregman iteration. J. Sci. Comput. **46**(1), 20–46 (2011)

Applied and Numerical Harmonic Analysis

(95 volumes)

1. A. I. Saichev and W. A. Woyczyński: *Distributions in the Physical and Engineering Sciences* (ISBN: 978-0-8176-3924-2)
2. C. E. D'Attellis and E. M. Fernandez-Berdaguer: *Wavelet Theory and Harmonic Analysis in Applied Sciences* (ISBN: 978-0-8176-3953-2)
3. H. G. Feichtinger and T. Strohmer: *Gabor Analysis and Algorithms* (ISBN: 978-0-8176-3959-4)
4. R. Tolimieri and M. An: *Time-Frequency Representations* (ISBN: 978-0-8176-3918-1)
5. T. M. Peters and J. C. Williams: *The Fourier Transform in Biomedical Engineering* (ISBN: 978-0-8176-3941-9)
6. G. T. Herman: *Geometry of Digital Spaces* (ISBN: 978-0-8176-3897-9)
7. A. Teolis: *Computational Signal Processing with Wavelets* (ISBN: 978-0-8176-3909-9)
8. J. Ramanathan: *Methods of Applied Fourier Analysis* (ISBN: 978-0-8176-3963-1)
9. J. M. Cooper: *Introduction to Partial Differential Equations with MATLAB* (ISBN: 978-0-8176-3967-9)
10. Procházka, N. G. Kingsbury, P. J. Payner, and J. Uhlir: *Signal Analysis and Prediction* (ISBN: 978-0-8176-4042-2)
11. W. Bray and C. Stanojevic: *Analysis of Divergence* (ISBN: 978-1-4612-7467-4)
12. G. T. Herman and A. Kuba: *Discrete Tomography* (ISBN: 978-0-8176-4101-6)
13. K. Gröchenig: *Foundations of Time-Frequency Analysis* (ISBN: 978-0-8176-4022-4)
14. L. Debnath: *Wavelet Transforms and Time-Frequency Signal Analysis* (ISBN: 978-0-8176-4104-7)
15. J. J. Benedetto and P. J. S. G. Ferreira: *Modern Sampling Theory* (ISBN: 978-0-8176-4023-1)

H. Boche et al. (eds.), *Compressed Sensing and Its Applications*,
Applied and Numerical Harmonic Analysis,
https://doi.org/10.1007/978-3-319-73074-5

16. D. F. Walnut: *An Introduction to Wavelet Analysis* (ISBN: 978-0-8176-3962-4)
17. A. Abbate, C. DeCusatis, and P. K. Das: *Wavelets and Subbands* (ISBN: 978-0-8176-4136-8)
18. O. Bratteli, P. Jorgensen, and B. Treadway: *Wavelets Through a Looking Glass* (ISBN: 978-0-8176-4280-80
19. H. G. Feichtinger and T. Strohmer: *Advances in Gabor Analysis* (ISBN: 978-0-8176-4239-6)
20. O. Christensen: *An Introduction to Frames and Riesz Bases* (ISBN: 978-0-8176-4295-2)
21. L. Debnath: *Wavelets and Signal Processing* (ISBN: 978-0-8176-4235-8)
22. G. Bi and Y. Zeng: *Transforms and Fast Algorithms for Signal Analysis and Representations* (ISBN: 978-0-8176-4279-2)
23. J. H. Davis: *Methods of Applied Mathematics with a MATLAB Overview* (ISBN: 978-0-8176-4331-7)
24. J. J. Benedetto and A. I. Zayed: *Sampling, Wavelets, and Tomography* (ISBN: 978-0-8176-4304-1)
25. E. Prestini: *The Evolution of Applied Harmonic Analysis* (ISBN: 978-0-8176-4125-2)
26. L. Brandolini, L. Colzani, A. Iosevich, and G. Travaglini: *Fourier Analysis and Convexity* (ISBN: 978-0-8176-3263-2)
27. W. Freeden and V. Michel: *Multiscale Potential Theory* (ISBN: 978-0-8176-4105-4)
28. O. Christensen and K. L. Christensen: *Approximation Theory* (ISBN: 978-0-8176-3600-5)
29. O. Calin and D.-C. Chang: *Geometric Mechanics on Riemannian Manifolds* (ISBN: 978-0-8176-4354-6)
30. J. A. Hogan: *Time?Frequency and Time?Scale Methods* (ISBN: 978-0-8176-4276-1)
31. C. Heil: *Harmonic Analysis and Applications* (ISBN: 978-0-8176-3778-1)
32. K. Borre, D. M. Akos, N. Bertelsen, P. Rinder, and S. H. Jensen: *A Software-Defined GPS and Galileo Receiver* (ISBN: 978-0-8176-4390-4)
33. T. Qian, M. I. Vai, and Y. Xu: *Wavelet Analysis and Applications* (ISBN: 978-3-7643-7777-9)
34. G. T. Herman and A. Kuba: *Advances in Discrete Tomography and Its Applications* (ISBN: 978-0-8176-3614-2)
35. M. C. Fu, R. A. Jarrow, J.-Y. Yen, and R. J. Elliott: *Advances in Mathematical Finance* (ISBN: 978-0-8176-4544-1)
36. O. Christensen: *Frames and Bases* (ISBN: 978-0-8176-4677-6)
37. P. E. T. Jorgensen, J. D. Merrill, and J. A. Packer: *Representations, Wavelets, and Frames* (ISBN: 978-0-8176-4682-0)
38. M. An, A. K. Brodzik, and R. Tolimieri: *Ideal Sequence Design in Time-Frequency Space* (ISBN: 978-0-8176-4737-7)
39. S. G. Krantz: *Explorations in Harmonic Analysis* (ISBN: 978-0-8176-4668-4)
40. B. Luong: *Fourier Analysis on Finite Abelian Groups* (ISBN: 978-0-8176-4915-9)

41. G. S. Chirikjian: *Stochastic Models, Information Theory, and Lie Groups, Volume 1* (ISBN: 978-0-8176-4802-2)
42. C. Cabrelli and J. L. Torrea: *Recent Developments in Real and Harmonic Analysis* (ISBN: 978-0-8176-4531-1)
43. M. V. Wickerhauser: *Mathematics for Multimedia* (ISBN: 978-0-8176-4879-4)
44. B. Forster, P. Massopust, O. Christensen, K. Gröchenig, D. Labate, P. Vandergheynst, G. Weiss, and Y. Wiaux: *Four Short Courses on Harmonic Analysis* (ISBN: 978-0-8176-4890-9)
45. O. Christensen: *Functions, Spaces, and Expansions* (ISBN: 978-0-8176-4979-1)
46. J. Barral and S. Seuret: *Recent Developments in Fractals and Related Fields* (ISBN: 978-0-8176-4887-9)
47. O. Calin, D.-C. Chang, and K. Furutani, and C. Iwasaki: *Heat Kernels for Elliptic and Sub-elliptic Operators* (ISBN: 978-0-8176-4994-4)
48. C. Heil: *A Basis Theory Primer* (ISBN: 978-0-8176-4686-8)
49. J. R. Klauder: *A Modern Approach to Functional Integration* (ISBN: 978-0-8176-4790-2)
50. J. Cohen and A. I. Zayed: *Wavelets and Multiscale Analysis* (ISBN: 978-0-8176-8094-7)
51. D. Joyner and J.-L. Kim: *Selected Unsolved Problems in Coding Theory* (ISBN: 978-0-8176-8255-2)
52. G. S. Chirikjian: *Stochastic Models, Information Theory, and Lie Groups, Volume 2* (ISBN: 978-0-8176-4943-2)
53. J. A. Hogan and J. D. Lakey: *Duration and Bandwidth Limiting* (ISBN: 978-0-8176-8306-1)
54. G. Kutyniok and D. Labate: *Shearlets* (ISBN: 978-0-8176-8315-3)
55. P. G. Casazza and P. Kutyniok: *Finite Frames* (ISBN: 978-0-8176-8372-6)
56. V. Michel: *Lectures on Constructive Approximation* (ISBN : 978-0-8176-8402-0)
57. D. Mitrea, I. Mitrea, M. Mitrea, and S. Monniaux: *Groupoid Metrization Theory* (ISBN: 978-0-8176-8396-2)
58. T. D. Andrews, R. Balan, J. J. Benedetto, W. Czaja, and K. A. Okoudjou: *Excursions in Harmonic Analysis, Volume 1* (ISBN: 978-0-8176-8375-7)
59. T. D. Andrews, R. Balan, J. J. Benedetto, W. Czaja, and K. A. Okoudjou: *Excursions in Harmonic Analysis, Volume 2* (ISBN: 978-0-8176-8378-8)
60. D. V. Cruz-Uribe and A. Fiorenza: *Variable Lebesgue Spaces* (ISBN: 978-3-0348-0547-6)
61. W. Freeden and M. Gutting: *Special Functions of Mathematical (Geo-)Physics* (ISBN: 978-3-0348-0562-9)
62. A. I. Saichev and W. A. Woyczyński: *Distributions in the Physical and Engineering Sciences, Volume 2: Linear and Nonlinear Dynamics of Continuous Media* (ISBN: 978-0-8176-3942-6)
63. S. Foucart and H. Rauhut: *A Mathematical Introduction to Compressive Sensing* (ISBN: 978-0-8176-4947-0)
64. G. T. Herman and J. Frank: *Computational Methods for Three-Dimensional Microscopy Reconstruction* (ISBN: 978-1-4614-9520-8)

65. A. Paprotny and M. Thess: *Realtime Data Mining: Self-Learning Techniques for Recommendation Engines* (ISBN: 978-3-319-01320-6)
66. A. I. Zayed and G. Schmeisser: *New Perspectives on Approximation and Sampling Theory: Festschrift in Honor of Paul Butzer's 85*th *Birthday* (ISBN: 978-3-319-08800-6)
67. R. Balan, M. Begue, J. Benedetto, W. Czaja, and K. A. Okoudjou: *Excursions in Harmonic Analysis, Volume 3* (ISBN: 978-3-319-13229-7)
68. H. Boche, R. Calderbank, G. Kutyniok, and J. Vybiral: *Compressed Sensing and its Applications* (ISBN: 978-3-319-16041-2)
69. S. Dahlke, F. De Mari, P. Grohs, and D. Labate: *Harmonic and Applied Analysis: From Groups to Signals* (ISBN: 978-3-319-18862-1)
70. A. Aldroubi: *New Trends in Applied Harmonic Analysis* (ISBN: 978-3-319-27871-1)
71. M. Ruzhansky: *Methods of Fourier Analysis and Approximation Theory* (ISBN: 978-3-319-27465-2)
72. G. Pfander: *Sampling Theory, a Renaissance* (ISBN: 978-3-319-19748-7)
73. R. Balan, M. Begue, J. Benedetto, W. Czaja, and K. A. Okoudjou: *Excursions in Harmonic Analysis, Volume 4* (ISBN: 978-3-319-20187-0)
74. O. Christensen: *An Introduction to Frames and Riesz Bases, Second Edition* (ISBN: 978-3-319-25611-5)
75. E. Prestini: *The Evolution of Applied Harmonic Analysis: Models of the Real World, Second Edition* (ISBN: 978-1-4899-7987-2)
76. J. H. Davis: *Methods of Applied Mathematics with a Software Overview, Second Edition* (ISBN: 978-3-319-43369-1)
77. M. Gilman, E. M. Smith, and S. M. Tsynkov: *Transionospheric Synthetic Aperture Imaging* (ISBN: 978-3-319-52125-1)
78. S. Chanillo, B. Franchi, G. Lu, C. Perez, and E. T. Sawyer: *Harmonic Analysis, Partial Differential Equations and Applications* (ISBN: 978-3-319-52741-3)
79. R. Balan, J. Benedetto, W. Czaja, M. Dellatorre, and K. A. Okoudjou: *Excursions in Harmonic Analysis, Volume 5* (ISBN: 978-3-319-54710-7)
80. I. Pesenson, Q. T. Le Gia, A. Mayeli, H. Mhaskar, and D. X. Zhou: *Frames and Other Bases in Abstract and Function Spaces: Novel Methods in Harmonic Analysis, Volume 1* (ISBN: 978-3-319-55549-2)
81. I. Pesenson, Q. T. Le Gia, A. Mayeli, H. Mhaskar, and D. X. Zhou: *Recent Applications of Harmonic Analysis to Function Spaces, Differential Equations, and Data Science: Novel Methods in Harmonic Analysis, Volume 2* (ISBN: 978-3-319-55555-3)
82. F. Weisz: *Convergence and Summability of Fourier Transforms and Hardy Spaces* (ISBN: 978-3-319-56813-3)
83. C. Heil: *Metrics, Norms, Inner Products, and Operator Theory* (ISBN: 978-3-319-65321-1)
84. S. Waldron: *An Introduction to Finite Tight Frames: Theory and Applications.* (ISBN: 978-0-8176-4814-5)
85. D. Joyner and C. G. Melles: *Adventures in Graph Theory: A Bridge to Advanced Mathematics.* (ISBN: 978-3-319-68381-2)

86. B. Han: *Framelets and Wavelets: Algorithms, Analysis, and Applications* (ISBN: 978-3-319-68529-8)
87. H. Boche, G. Caire, R. Calderbank, M. März, G. Kutyniok, and R. Mathar: *Compressed Sensing and Its Applications* (ISBN: 978-3-319-69801-4)
88. A. I. Saichev and W. A. Woyczyński: *Distributions in the Physical and Engineering Sciences, Volume 3: Random and Fractal Signals and Fields* (ISBN: 978-3-319-92584-4)
89. G. Plonka, D. Potts, G. Steidl, and M. Tasche: *Numerical Fourier Analysis* (978-3-030-04305-6)
90. K. Bredies and D. Lorenz: *Mathematical Image Processing* (ISBN: 978-3-030-01457-5)
91. H. G. Feichtinger, P. Boggiatto, E. Cordero, M. de Gosson, F. Nicola, A. Oliaro, and A. Tabacco: *Landscapes of Time-Frequency Analysis* (ISBN: 978-3-030-05209-6)
92. E. Liflyand: *Functions of Bounded Variation and Their Fourier Transforms* (978-3-030-04428-2)
93. R. Campos: *The XFT Quadrature in Discrete Fourier Analysis* (978-3-030-13422-8)
94. M. Abell, E. Iacob, A. Stokolos, S. Taylor, S. Tikhonov, J. Zhu: *Topics in Classical and Modern Analysis: In Memory of Yingkang Hu* (978-3-030-12276-8)
95. H. Boche, G. Caire, R. Calderbank, G. Kutyniok, R. Mathar, P. Petersen: *Compressed Sensing and its Applications: Third International MATHEON Conference 2017* (978-3-319-73073-8)

For an up-to-date list of ANHA titles, please visit
http://www.springer.com/series/4968