High-Dimensional Data Analysis with Low-Dimensional Models

Theory, Algorithms, and Applications

John Wright (Columbia University)
Yi Ma (University of California, Berkeley)
Allen Y. Yang (University of California, Berkeley)

August 15, 2018

Copyright ©2014 Reserved
No parts of this draft may be reproduced or distributed without written permission
from the authors.

Preface

The Era of Big Data

In the past decade or so, our world has entered the age of "Big Data." The information technological industry is now facing the challenge of processing and analyzing massive amounts of data on a daily basis. The size and the dimension of the data have reached unprecedented scale and are still increasing at unprecedented rate.

For instance, on the technological side, the resolution of consumer digital cameras has increased nearly ten-fold in the past decade or so. Each day, over 300 million photos are uploaded to Facebook;[1] 300 hours of videos are posted on Youtube every minute; and over 20 million entertaining short videos are produced and posted to Toutiao of China.[2]

On the business side, on a single busy day, Alibaba.com needs to take in over 800 million purchase orders for over 15 million products, handles over a billion payments, and delivers more than 30 million packages. Those numbers are still growing and growing fast!

On the scientific front, super-resolution microscopic imaging technologies, such as the scanning tunneling microscope (STM), have undergone tremendous advances in the past decades, and are now capable of producing images with subatomic resolution. High-throughput gene sequencing

[1] Almost all of them are passing through several processing pipelines for face detection, face recognition, and general object classification, etc.

[2] The largest Internet media platform in China, and all the videos need to be scrutinized for bad content or for proper recommendation.

Figure 1. Two images of the same person, the resolution of the image on the left is 1200×1800 (300 pixels per inch), whereas the image on the right is down-sampled to 120×180 (30 pixels per inch), with only 1/100th fraction of pixels of the original one.

technologies are capable of sequencing hundreds of millions of DNA molecules at a time, and can sequence in just a few hours an entire human genome that has a length of over 3 billion base pairs and contains 20,000 protein-encoding genes!

Paradigm Shift in Data Acquisition, Processing, and Analysis

In the past, scientists or engineers usually have control of the data acquisition apparatus and process. Since the apparatus is expensive and the process time-consuming, typically only sufficient data (or signals) are collected for a specific given task. The data or signals collected are mostly informative for the task and do not contain much redundant or irrelevant information. Hence, classical signal processing or data analysis typically operates under the premise that

$$\text{Classical Premise:} \quad \textbf{Data} \approx \textbf{Information},$$

and hence in the classical data processing paradigms one mostly needs to deal with problems such as removing noise or compressing the data for storage or transport.

As we have mentioned above, technologies such as the Internet, smart phones, and high-throughput imaging and gene sequencing have fundamentally changed the nature of data acquisition and analysis. We are moving from a "data-poor" era to a "data-rich" era. Nevertheless, data-rich does not necessarily imply "information-rich." Massive amounts of data are being collected even without any specific purposes in advance. Scientists or engineers often do not have direct control of the data acquisition process, neither in its quantity nor in quality. Therefore, any given new task could be inundated with massive data that are very redundant or irrelevant to the task at hand.

To see intuitively why this is the case, let us first consider the problem of *face recognition*. Figure 1 shows two images of the same person. It is arguably the case that to human eyes, both images convey the identity of

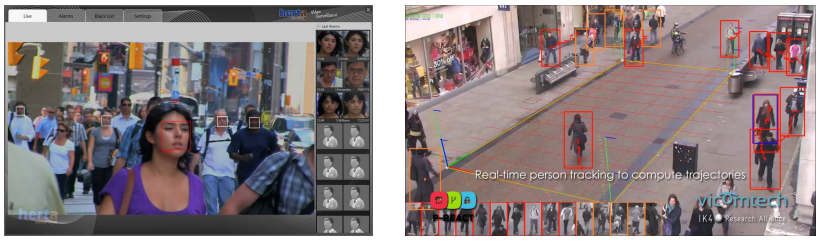

Figure 2. Left: face detection; Right: pedestrian detection in surveillance videos.

the person equally well, even though the resolution of the second image is merely 1/100th of the first one. In other words, if we view both images as vectors with their pixel values as coordinates, then the dimension of the low-resolution image vector is merely 1/100th of the original one. Clearly, the information about the identity of a person relies on statistics of much lower dimension than the original high-resolution image[3]. Hence, in such scenarios, we have a new premise:

$$\text{New Premise I:} \quad \textbf{Data} \gg \textbf{Information}.$$

For *object detection* tasks (such as face detection or pedestrian detection in surveillance videos), the issue is no longer with redundancy. Instead, the difficulty is to find any relevant information at all in an ocean of irrelevant data. In typical surveillance videos as shown in Figure 2, image pixels associated with human faces or bodies only occupy a very tiny portion of the frame whereas majority of the pixels belong to completely irrelevant objects in the scene. In addition, the subject of interest might present only in a very short segment of the entire video. The same type of "detection" tasks also arise in studying genetics: Out of the nearly 20,000 protein-encoding genes, scientists need to identify which one (or handful ones) are responsible for certain genetic disease. In scenarios like these, we have:

$$\text{New Premise II:} \quad \textbf{Data} = \textbf{Information} + \textbf{Irrelevant Data}.$$

The explosive growth of e-commerce, online shopping, social networks has created tremendous datasets of user preferences. Major internet companies typically have records of billions of people's preferences in millions of commercial products, media contents, and more. By nature, such datasets of user preferences, however massive, are far from complete. For instance, in the case of a dataset of movie ratings as shown in Figure 3, no one

[3]In fact, one can continue to argue that even the low-resolution image is still highly redundant. Studies have shown that humans can recognize familiar faces from images with a resolution as low as around 18×18 pixels. Modern face recognition algorithms extract merely a few hundred of features for reliable face verification.

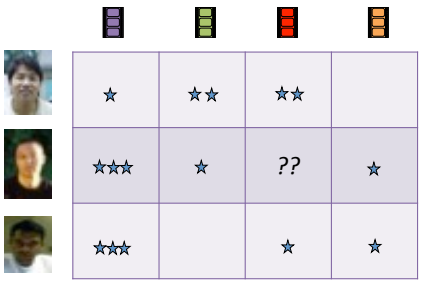

Figure 3. An example of collaborative filtering of user preferences: how to guess a customer's rating for a movie even if he or she has not seen it yet?

could have seen all the movies and no movie would have been seen by all people. Nevertheless, companies like Netflix need to guess from such incomplete datasets a customer's preference in any given movie so that they could send the most relevant recommendations or advertisements to the customer. This problem in information retrieval literature is known as *collaborative filtering*, and most internet companies' business[4] relies on solving this problem effectively and efficiently. The most fundamental reason why complete information can be derived from such a highly-incomplete dataset is because user preferences are not random and such data have good structures. For instance, many people have similar tastes in movies and many movies are similar in styles. Rows and columns of the user preference table would be strongly correlated, hence the intrinsic dimension (or rank) of the complete table is in fact extremely low compared to its size. For large (incomplete) datasets with good low-dimensional structures, we have:

New Premise III: **Incomplete Data $\approx$ Complete Information**.

As above examples have indicated, in the modern era of big data, we often face problems of recovering specific information that is buried in highly redundant, irrelevant, or seemingly incomplete data sets. Such information is typically encoded as certain low-dimensional structures of the data or only depends on a small (or sparse) subset of a massive dataset. This is very different from the classical setting and is precisely the reason why modern data science and engineering are undergoing a fundamental shift in their mathematical and computational paradigms. At its foundation, we need to develop a new mathematical theory that characterizes precise conditions under which such low-dimensional information can be correctly

[4]Most internet companies make money from advertisements, including but not limited to Google, Facebook, Amazon, Alibaba, etc.

and effectively recovered. Equally importantly, we need to develop truly efficient and scalable algorithms that are capable of extracting such information from massive high-dimensional datasets at unprecedented speed and arbitrary scale.

A Universal Task: Pursuit of Low-dimensionality

The problem of identifying low-dimensional structures in high-dimensional spaces is actually one of the most fundamental problems that interweave through a long history many related fields such as systems theory, statistical inference, signal processing, supervised learning, and optimization.

- **Low-dimensionality of physical signals.** The low-dimensionality of real-world signals or data often arises from the intrinsic physical mechanisms from which the data are generated. Many real-world signals or data are observations of physical processes governed by certain physical generative mechanisms. For instance, magnetic resonance (MR) images are generated by manipulating magnetic fields that obey the Maxwell's equations; dynamics of any mechanical systems (such as cars and robots) follow Newton's laws of motion. In systems theory, such dynamics can often be modeled or approximately modeled by a set of (linear) differential equations, also known as a *state-space model*:

$$\begin{cases} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t), \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{z}(t), \end{cases} \tag{0.0.1}$$

  where $\boldsymbol{x} \in \mathbb{R}^n$ is the state, $\boldsymbol{y} \in \mathbb{R}^{n_o}$ is the output, and $\boldsymbol{u} \in \mathbb{R}^{n_i}$ is the input. According to the theory of system identification [Van Overschee and de Moor, 1996], the observed output $\boldsymbol{Y} = \{\boldsymbol{y}(t)\}_{t=1}^{\infty}$ as a function in time t always lives a subspace of dimension no more than $n = \dim(\boldsymbol{x})$.[5]. Identifying this n-dimensional subspace associated with the output $\boldsymbol{Y}$ is the key to identify the parameters of the system $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C})$. Notice that, in modern practice of *deep neural networks* (DNNs), variants to such state-space models[6] have been widely adopted, also known as *recursive neural networks* (RNNs), for modeling time-series data such as speech signals and videos [**?**]. Of course, there are many other physical sources for low-dimensional structures that do not necessarily involve dynamics. Regardless, this book will show how to impose general low-dimensional structures for efficient and effective identification of such physical models.

[5] more precisely speaking, the so-called Hankel matrix associated with the output $\boldsymbol{Y}$ always has rank less than or equal to n, regardless of its dimension or length.

[6] usually with additional nonlinear activations introduced to places in the state space model.

- **Low-dimensionality for modeling data correlation.** In the practice of modern data science, we often deal with data that are not necessarily generated from any physical processes or whose generative mechanisms remain unclear. The aforementioned user preference data, web documents, natural languages, gene expression data are some examples. Nevertheless, such data are by no means structureless, and there is usually strong and rich statistical correlation or dependency among the data. To model such correlation, one may view the observed data as samples of a set of random variables $\boldsymbol{x}_o$, and the correlation among the observed data can be modeled through their conditional probability on another set of hidden *latent* variables $\boldsymbol{x}_h$. The final map of dependencies among random variables $\boldsymbol{x} = (\boldsymbol{x}_o, \boldsymbol{x}_h)$ is often called a *graphical model* [Jordan, 2003], usually denoted as $\mathcal{G} = (V, E)$.[7] For instance, the above state-space model (0.0.1) can be considered as a special case of such latent variable graphical models[8]. A fundamental and challenging problem in statistical learning is how to infer the latent variables and their relationships only from (marginal) statistics of the observed ones even if the number of latent variables and their relationships are all unknown. In the most basic case when all the variables are jointly Gaussian, it has been shown that such a graphical model is identifiable only if the graphical model $\mathcal{G}$ is sufficiently sparse [Chandrasekaran et al., 2012]. Trees and multi-layer deep networks are representative examples. For such graphs, the covariance matrix $\boldsymbol{\Sigma}_o$ of the observed variables $\boldsymbol{x}_o$ always has the following decomposable structure:

$$\boldsymbol{\Sigma}_o^{-1} = \boldsymbol{L} + \boldsymbol{S}, \tag{0.0.2}$$

  where $\boldsymbol{S}$ is a sparse matrix and $\boldsymbol{L}$ is a low-rank matrix. The rank of $\boldsymbol{L}$ is associated with the number of independent latent variables in the graph: $\operatorname{rank}(\boldsymbol{L}) = \dim(\boldsymbol{x}_h)$; the sparse matrix $\boldsymbol{S}$ is associated with the conditional statistics of the observed variables given the latent ones – an entry s_{ij} of $\boldsymbol{S}$ is zero if the two observed variables $\boldsymbol{x}_o(i)$ and $\boldsymbol{x}_o(j)$ are conditionally independent given the latent variables $\boldsymbol{x}_h$. Decomposing the observation $\boldsymbol{\Sigma}_o^{-1}$ into a low-rank matrix $\boldsymbol{L}$ and a sparse matrix $\boldsymbol{S}$ is then crucial to the problem of inferring the full graphical model $\mathcal{G}$. Although the decomposition problem (0.0.2) is generally *NP-hard*, we will see in this book how to solve it correctly and efficiently by leveraging the low-dimensionality of $\boldsymbol{L}$ and $\boldsymbol{S}$.

[7] The set of vertices V consists of all the random variables $V = \{\boldsymbol{x}_o, \boldsymbol{x}_h\}$, and the set of edges E indicate dependency among pairs of random variables $\mathbb{P}[\boldsymbol{x}(i) \mid \boldsymbol{x}(j)]$.

[8] the output $\boldsymbol{y}$ would be the observations and the state $\boldsymbol{x}$ would be the hidden latent variables.

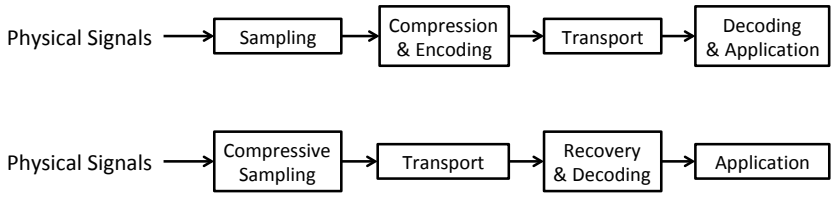

Figure 4. Comparison of classical signal acquisition and processing pipeline (top) and the compressive sensing pipeline (bottom).

- **Harnessing low-dimensionality for efficient data acquisition.** In classical signal processing, the intrinsic low-dimensionality of data is mostly exploited for purposes of efficient storage and transport. Figure 4 top illustrates a traditional pipeline for data acquisition and processing: A physical (continuous) signal is first acquired by sampling at a rate twice its maximal frequency – according to the classic *Nyquist-Shannon* sampling theorem [Oppenheim et al., 1999]. The so-sampled (and digitized) signal is then compressed based on its intrinsic low-dimensional structures. For instance, piecewise smooth signals can be represented by relatively few number of wavelet bases, which serves as the basis for the popular image compression scheme JPEG 2000. The so-compressed signals are then used for storage, transport, and decoded later for various applications. However, for physical signals like images that contain very high-frequency components, sampling at the Nyquist rate can be extremely challenging at the sensor end. For instance, in order to capture ever sharper edges in an image, the number of pixels of the imaging sensor in our digital cameras has increased dramatically in recent years. Such a brute force sampling scheme is obviously rather wasteful since sharp edges occupy only very tiny fraction of the image and yet all the relatively smooth regions are sampled at the same rate. In cases such as medical imaging, such brute force increasing of sampling density is not even allowed due to patient comfort and safety. As it turns out, the number of samples truly needed to recover a signal should be (almost) proportional to the intrinsic dimension of the signal, which can be significantly lower than the Nyquist rate [Tropp, 2010, Mishali and Eldar, 2010]! At the acquisition front, one only needs to keep a much fewer number of informative samples of the signal, hence the notion of "*compressive sampling*" or "*compressive sensing*" [Donoho et al., 2006, Candès and Tao, 2006, Candès, 2006]. Such compressive samples are already efficient for storage and transport, and the complete signal can be recovered later when it is actually used. See Figure 4 bottom for an illustration of this new data acquisition and processing

paradigm, which has revolutionized the field of signal processing in the past decade. In this book, we will systematically study the theoretical basis for designing such informative samples and developing algorithms for recovering the full signal from such samples effectively and efficiently. We will also see a few striking applications of this new paradigm at work.

- **Scalable algorithms for low-dimensional information.** As we have argued before, many modern data science problems require us to effective retrieve relevant (low-dimensional) information from very high-dimensional data with tremendous redundancy and irrelevant data. The resulting systems need to process data of dimension and size in the millions or even billions and execute algorithms that solve the associated optimization problems of the same size. Hence, such algorithms must be extremely efficient and scalable. In this book, based on principled methods and techniques from optimization, we will demonstrate how to systematically derive optimal algorithms for recovering broad classes of low-dimensional structures with increasing efficiency and scalability. It has been observed that the iterative structures of such algorithms bear many resemblances to structures discovered in modern deep neural networks. Recent studies have started to reveal that, from an information-theoretic perspective [**?**], deep neural networks (DNNs) trained for tasks such as image classification[9] are essentially trying to find a low-rank approximation to the joint distribution of the input $\boldsymbol{X}$ and output $\boldsymbol{Y}$:

$$\min_{\boldsymbol{\Phi},\boldsymbol{\Psi}} \|\boldsymbol{D} - \boldsymbol{\Phi}\boldsymbol{\Psi}^*\|_2, \tag{0.0.3}$$

  where $\boldsymbol{D} \in \mathbb{R}^{m\times n}$ is (a normalized version of) the joint distribution (matrix) $\mathbb{P}[\boldsymbol{X},\boldsymbol{Y}]$,[10] and $\boldsymbol{\Phi} \in \mathbb{R}^{m\times k}$ and $\boldsymbol{\Psi} \in \mathbb{R}^{n\times k}$ are two factor matrices of rank k. This should come at no surprise since such neural networks are precisely trained to find the correlation between the input and output, and the principal components of the joint distribution matrix best capture such correlation. Hence, on one hand, one could view modern neural networks are flexible extensions to these principled algorithms for recovering wider range of low-dimensional structures under more flexible conditions and assumptions; and on the other hand, through a systematic study of the principled methods and algorithms for low-dimensional structures, one could potentially gain useful insights for developing new network architectures that could be more effective and efficient in learning low-dimensional structures (from high-dimensional data).

[9] typically using the so-called "soft mask" function as the loss function.

[10] Here for simplicity, but without loss of generality, we consider the input $\boldsymbol{X}$ and output $\boldsymbol{Y}$ as discrete random variables of m and n states, respectively.

Purposes of This Book

Over more than a decade, there have been explosive developments in the study of low-dimensional structures in high-dimensional spaces. To a large extent, the geometric and statistical properties of representative low-dimensional models (such as sparse and low-rank) are now well understood. Conditions under which such models can be effectively and efficiently recovered from (minimal measurement) data have been clearly characterized. Many highly efficiently and scalable algorithms have been developed for recovering such low-dimensional models from high-dimensional data. Their working conditions and convergence rates are also thoroughly characterized. These new theoretical results and algorithms have revolutionized the practice of signal processing, and have had significant impacts on sensing, imaging, statistical learning, and data science. They have significantly advanced the state of the art for many applications such as compressive sensing[11], image processing[12], computer vision[13], bioinformatics[14], and information retrieval[15]. As we will see in this book, some of the advancements are so remarkable that they simply defy conventional wisdom.

As witnesses to such historical advancements, we believe that the time is now ripe to give a comprehensive survey of this new body of knowledge and to organize these rich results under a unified theoretical and computational framework. There are a number of excellent existing books on this topic that already focus on the mathematical theory of compressive sensing and sparse models [Foucart and Rauhut, ,Hastie et al., 2015,Van De Geer, 2016]. Nevertheless, the goal of this book is to bridge an apparent gap between theory, computation, and practice of a broad family representative low-dimensional structures (including sparse and low-rank models, and their extensions and variants). Not only does the book establish a unified and complete mathematical theory for effectively recovering such low-dimensional structures, but it also shows how to systematically develop efficient and scalable algorithms for solving the recovery problems using advanced optimization principles and techniques. Furthermore, through diverse and rich tasks in science and engineering, the book further coaches how to correctly incorporate domain knowledge in order to successfully apply these new models and algorithms to solve real-world problems. As a result, we believe the book has good values for both theoreticians and practitioners in related fields.

[11] compressive sampling and recovery of medical and microscopic images, etc.
[12] denoising and super-resolution of natural images, etc.
[13] regular texture synthesis, camera calibration, and 3D reconstruction, etc.
[14] microarray data analysis for gene-protein relations etc.
[15] collaborative filtering of documents and multimedia data etc.

Intended Audience of This Book

In many ways, the body of knowledge covered in this book has great pedagogical values. Through rigorous and elegant mathematical development, we hope our readers are able to gain remarkable new knowledge and insights about high-dimensional geometry and statistics, far beyond what has been established in classical signal processing and data analysis (for compression or other purposes). Such insights are applicable and generalizable to a wide range of useful low-dimensional structures and models, and can lead to entirely new methods and algorithms that revolutionize solutions to many important scientific and engineering problems.

Hence, this book is intended to be a textbook for a course that introduces basic mathematical and computational principles for sensing, processing, and analyzing high-dimensional data with low-dimensional structures. The *targeted core audience* of this book are senior undergraduate and entry-level graduate students in Electrical Engineering and/or Computer Science, especially in the areas of signal processing, data science, optimization, machine learning, and their applications. This book helps the students to receive systematic and rigorous training in concepts and methods of high-dimensional geometry, statistics, and optimization. Through a very diverse and rich set of applications and exercises, the book also coaches students how to correctly use such concepts and methods to model real-world data and solve real-world engineering and scientific problems.

The book is written to be friendly to instructors and students. It provides ample illustrations, examples, exercises, and programs from which students may gain hands-on experience with the concepts and methods covered in the book. We have taught several times a one-semester graduate course based on material from this book at the University of Illinois at Urbana-Champaign, Columbia University, ShanghaiTech University, and the University of California at Berkeley. The main prerequisites for the course are college-level linear algebra and probability. To make this book accessible to a broader audience, we have tried to make the book as self-contained as possible: We give a crisp summary of facts used in this book from linear algebra, optimization, and statistics in the Appendices. For EECS students, preliminary courses on signal processing, matrix analysis, and optimization will improve their appreciation. From our experiences, besides beginning graduate students, many senior undergraduate students at these institutes were able to take the course and read the book without much difficulty.

Organization of This Book

We divide the material of this book into four Parts:

- *Part I: Basic Theory* develops the fundamental theoretical results for sparse, low-rank, and general low-dimensional models. It characterizes the conditions under which the the inverse problems of recovering such low-dimensional structures become trackable and can be solved efficiently, with guaranteed accuracy.
- *Part II: Fast and Scalable Algorithms* introduces techniques from optimization to develop practical algorithms for recovering the low-dimensional models that are fast and scalable to large data size and high dimension.
- *Part III: Applications* demonstrates how methods in this book could significantly improve the solutions to a variety of real-world applications. The applications also coach how the general models and algorithms introduced in this book should be properly customized and extended to incorporate additional domain knowledge (priors or constraints) about the applications.
- *Part IV: Appendices* at the end of the book are meant to make the book self-contained. They cover basic mathematical concepts and results from Linear Algebra, Optimization, and High-Dimensional Statistics that are used in the main body of the book.

How to Use This Book to Teach or Learn

The book contains enough material for a two-semester course series. We have purposely organized the material in the book in a modular fashion so that the chapters and even sections can be easily rearranged to support different types of courses. Here are some examples:

- *A Short Course on Sparsity*: The two theory Chapters 2 and 3; the convex optimization Chapter 8, and the three application Chapters 11–13, plus some appendices will be adequate for a six to eight-week short (summer) course for senior undergraduate students and early year graduate students.
- *A One-Semester Course on Low-Dimensional Models*: The four theory Chapters 2–5; the convex optimization Chapter 8, and the several application Chapters 11–15, plus the appendices will be adequate for a one-semester course on low-dimensional models for graduate students.
- *An Advanced Topic Course on High-Dimensional Data Analysis*: With the previous course as prerequisite, the two theory Chapters 6 and 7 on general and nonlinear models; the algorithm chapter on nonlinear optimization, extensions to deep neural networks, the more

advanced and broader applications in Chapters 16 and 17, plus new advances of the field selected from recent literature would make a good second and more advanced course for graduate students who are interested in doing research in this area.

John Wright, New York
Yi Ma, Berkeley, California
Allen Y. Yang, Berkeley, California
August 15, 2018

Contents

Preface **iii**

List of Symbols **xxiii**

1 Introduction **xxvii**
1.1 Dealing High-Dimensional Data xxvii
1.2 A Brief History xxvii
1.3 The Modern Era 1
1.4 This Book 1

I Basic Theory **3**

2 Sparse Signal Models **5**
2.1 Applications of Sparse Signal Modeling 5
2.1.1 An Example from Medical Imaging 6
2.1.2 An Example from Image Processing 10
2.1.3 An Example from Face Recognition 12
2.2 Recovering a Sparse Solution 14
2.2.1 Norms on Vector Spaces 14
2.2.2 The ℓ^0 Norm 16
2.2.3 The Sparsest Solution: Minimizing the ℓ^0 Norm . 17
2.2.4 Computational Complexity of ℓ^0 Minimization . . 21
2.3 Relaxing the Sparse Recovery Problem 23

2.3.1 Convex Functions 23
2.3.2 A Convex Surrogate for the ℓ^0 Norm: the ℓ^1 Norm 26
2.3.3 A Simple Test of ℓ^1 Minimization 27
2.4 Summary 34
2.5 Notes and References 34
2.A Complexity Classes and NP-Hardness 40

3 Convex Methods for Sparse Signal Recovery 42
3.1 *Why* Does ℓ^1 Minimization Succeed? Geometric Intuitions 43
3.2 A First Correctness Result for Incoherent Matrices . . . 45
3.2.1 Coherence of a Matrix 46
3.2.2 Correctness of ℓ^1 Minimization 48
3.2.3 Constructing an Incoherent Matrix 52
3.3 Towards Stronger Correctness Results 53
3.3.1 Limitations of Incoherence 53
3.3.2 The Restricted Isometry Property (RIP) 56
3.3.3 Restricted Strong Convexity Condition 59
3.3.4 Success of ℓ^1 Minimization under RIP 63
3.4 Matrices with Restricted Isometry Property 65
3.4.1 The Johnson-Lindenstrauss Lemma 66
3.4.2 RIP of Gaussian Matrices 68
3.4.3 RIP of Non-Gaussian Matrices 72
3.5 Noisy Observations or Approximate Sparsity 75
3.5.1 Stable Recovery of Sparse Signals 76
3.5.2 Recovery of Inexact Sparse Signals 84
3.6 Phase Transitions in Sparse Recovery 87
3.6.1 Phase Transitions: Main Conclusions 88
3.6.2 Phase Transitions via Coefficient-Space Geometry 91
3.6.3 Phase Transitions via Observation-Space Geometry 98
3.6.4 Phase Transitions in Support Recovery 100
3.7 Notes and References 108

4 Convex Methods for Low-Rank Matrix Recovery 109
4.1 Low-Rank Matrix Models 109
4.1.1 Applications of Low-Rank Modeling 110
4.1.2 Rank and Singular Value Decomposition (SVD) . 114
4.1.3 Best Low-Rank Matrix Approximation 115
4.2 Recovering a Low-Rank Matrix 119
4.2.1 General Rank Minimization Problems 119
4.2.2 Convex Relaxation of Rank Minimization 120
4.2.3 Nuclear Norm as A Convex Envelope of Rank . . 123
4.2.4 Success of Nuclear Norm under Rank-RIP 124
4.2.5 Rank-RIP of Random Measurements 129
4.2.6 Noise, Inexact Low-Rank, and Phase Transition . 134
4.3 Low-Rank Matrix Completion 140

4.3.1 Nuclear Norm Minimization for Matrix Completion 141
4.3.2 Nuclear Norm Minimization via the Augmented Lagrangian Method 141
4.3.3 When Does Nuclear Norm Minimization Solve Matrix Completion? 145
4.3.4 Proving Correctness of Nuclear Norm Minimization 148
4.3.5 Stable Matrix Completion with Noise 160
4.4 Summary 163
4.5 Notes and References 163
4.A Connections of ℓ^0 and Rank Minimization, Matrix Norms 171
4.B Nuclear Norm Minimization as a Semidefinite Program . 173

5 Decomposing Low-rank and Sparse Matrices 177
5.1 Combining Sparse and Low-rank Structures 177
5.1.1 Applications of Robust PCA 178
5.2 Robust PCA via Principal Component Pursuit 180
5.2.1 Convex Relaxation for Sparse Low-Rank Separation 181
5.2.2 Solving PCP via Alternating Directions Method . 181
5.2.3 Numerical Simulations and Experiments of PCP . 183
5.3 Identifiability and Exact Recovery 190
5.3.1 Identifiability Conditions 190
5.3.2 Correctness of Principal Component Pursuit . . . 194
5.4 Noise Stability of Principal Component Pursuit 203
5.4.1 Proof of Theorem 5.4.1 205
5.4.2 Recovery Results for Random Noise 207
5.5 Compressive Principal Component Pursuit 207
5.6 Matrix Completion with Corrupted Entries 209
5.7 Summary 211
5.8 Notes and References 211

6 Sensing General Low-Dimensional Models 215
6.1 Concise Signal Models 216
6.1.1 Atomic sets 216
6.1.2 Other examples of atomic sets 216
6.1.3 General optimization problems for recovering simple signals 219
6.2 Geometry and Phase Transitions 221
6.3 Statistical Analysis and Noise Stability 222
6.4 Limitations of Convex Relaxation 222
6.5 Notes and References 223

7 Learning Sparsifying Dictionaries from Data 224

II Fast and Scalable Algorithms **225**

8 Convex Optimization for Structured Signal Recovery **227**
8.1 Challenges and Opportunities 228
8.2 Proximal Gradient Methods 231
8.2.1 Proximal Gradient for the Lasso 235
8.2.2 Proximal Gradient for Stable PCP 237
8.2.3 Convergence of Proximal Gradient 238
8.3 Accelerated Proximal Gradient Methods 241
8.3.1 APG for Basis Pursuit Denoising 244
8.3.2 APG for Stable Principal Component Pursuit . . 245
8.3.3 Convergence of APG 245
8.4 Augmented Lagrange Multipliers 248
8.4.1 ALM for Basis Pursuit 253
8.4.2 ALM for Principal Component Pursuit 253
8.4.3 Convergence of ALM 254
8.5 Alternating Direction Method of Multipliers 255
8.5.1 ADMM for Principal Component Pursuit 257
8.5.2 Convergence of ADMM 258
8.6 Frank-Wolfe Methods for Scalable Optimization 260
8.6.1 Convergence of Frank-Wolfe 263
8.6.2 Frank-Wolfe for Stable Matrix Completion 265
8.6.3 Connection to Greedy Methods for Sparsity . . . 266
8.7 Notes and References 270

9 Nonconvex Optimization for Nonlinear Problems **273**

III Applications **275**

10 Robust Face Recognition **277**
10.1 Introduction 277
10.2 Classification Based on Sparse Representation 279
10.3 Robustness to Occlusion or Corruption 281
10.4 Dense Error Correction with the Cross and Bouquet . . 287
10.5 Notes and References 288

11 Magnetic Resonance Imaging **290**
11.1 Introduction 290
11.2 Formation of MR Images 291
11.2.1 Basic Physics 291
11.2.2 Selective Excitation and Spatial Encoding 293
11.2.3 Sampling and Reconstruction 294
11.3 Sparsity and Compressive Sampling of MR Images . . . 296
11.3.1 Sparsity of MR Images 296

11.3.2 Compressive Sampling of MR Images 299
11.4 Algorithms for MR Image Recovery 302
11.5 Notes and References 307

12 Wideband Spectrum Sensing **310**
12.1 Introduction 310
12.1.1 Wideband Communications 310
12.1.2 Nyquist Sampling and Beyond 311
12.2 Wideband Interferer Detection 313
12.2.1 Conventional Scanning Approaches 314
12.2.2 Compressive Sensing in the Frequency Domain . 316
12.3 System Implementation and Performance 318
12.3.1 Quadrature Analog to Information Converter . . 318
12.3.2 A Prototype Circuit Implementation 320
12.3.3 Recent Developments in Hardware Implementation 325
12.4 Notes and References 325

13 Robust Photometric Stereo **326**
13.1 Introduction 326
13.2 Photometric Stereo via Low-Rank Matrix Recovery . . . 328
13.2.1 Lambertian Surface under Directional Lights . . . 328
13.2.2 Modeling Shadows and Specularities 330
13.3 Robust Matrix Completion Algorithm 334
13.4 Experimental Evaluation 335
13.4.1 Quantitative Evaluation with Synthetic Images . 336
13.4.2 Qualitative Evaluation with Real Images 341
13.5 Notes and References 342

14 Structured Texture Recovery **344**
14.1 Introduction 344
14.2 Low-rank Textures 345
14.3 Structured Texture Inpainting 347
14.4 Transform Invariant Low-rank Textures 352
14.4.1 Deformed and Corrupted Low-rank Textures . . . 352
14.4.2 The TILT Algorithm 354
14.5 Applications of TILT 357
14.5.1 Planar Low-rank Textures 358
14.5.2 Generalized Cylindrical Surfaces 359
14.5.3 Calibrating Camera Lens Distortion 362
14.6 Notes and References 368

15 Scanning Tunneling Microscopy **370**
15.1 Introduction 370
15.2 Data Model for STM 372
15.2.1 Sparse Blind Deconvolution 372

15.2.2 Breaking Symmetry 374
15.2.3 Geometry of Optima on the Sphere 375
15.3 A Two-Stage Algorithm 375
15.4 Experimental Results 375
15.5 Notes and References 375

16 Applications to Other High-Dimensional Data 376
16.1 Topic Models for Document Analysis 377
16.1.1 Background 377
16.1.2 A Joint Topic-Document Model 378
16.1.3 Results 380
16.1.4 Discussions and References 382
16.2 Optimal Codes for Channel Capacity 384
16.2.1 Background 384
16.2.2 Model 384
16.2.3 Results 384
16.2.4 Discussions and References 384
16.3 Music and Lyrics Separation 385
16.3.1 Background 385
16.3.2 Model 385
16.3.3 Results 385
16.3.4 Discussions and References 385
16.4 Microarray Data Analysis 386
16.4.1 Background 386
16.4.2 Model 386
16.4.3 Results 386
16.4.4 Discussions and References 386
16.5 Internet Anomaly Traffic Analysis 387
16.5.1 Background 387
16.5.2 Model 387
16.5.3 Results 387
16.5.4 Discussions and References 387
16.6 Robust System Identification 388
16.6.1 Background 388
16.6.2 Model 388
16.6.3 Results 388
16.6.4 Discussions and References 388
16.7 Learning Graphical Models 389
16.7.1 Background 389
16.7.2 Model 389
16.7.3 Results 389
16.7.4 Discussions and References 389
16.8 Low-Dimensional Models and Deep Networks 390
16.8.1 Background 390
16.8.2 Model 390

16.8.3 Results 390
16.8.4 Discussions and References 390

IV Appendices 391

A Facts from Linear Algebra and Matrix Analysis 393
A.1 Vector Spaces, Linear Independence, Bases and Dimension 394
A.2 Inner Products 396
A.3 Linear Transformations and Matrices 397
A.4 Matrix Groups 400
A.5 Subspaces Associated with a Matrix 401
A.6 Linear Systems of Equations 402
A.7 Eigenvectors and Eigenvalues 404
A.8 The Singular Value Decomposition (SVD) 407
A.9 Vector and Matrix Norms 409

B Convex Sets and Functions 415
B.1 Convex Sets 416
B.2 Convex Functions 417
B.3 Subdifferentials of Nonsmooth Convex Functions 421

C Optimization Problems and Optimality Conditions 425
C.1 Unconstrained Optimization 425
C.2 Constrained Optimization 427
C.3 Basic Duality Theory 428

D Methods for Optimization 431
D.1 Gradient Descent 432
D.2 Rates of Convergence and Acceleration 434
D.3 Block Coordinate Descent and Alternating Direction Method of Multipliers 439
D.4 Nonconvex Problems 440

E Facts from High-Dimensional Statistics and Geometry (John) 441
E.1 Proofs of Basic Concentration Results 442
E.2 Thresholds for ℓ^∞ Norm of a Gaussian Vector. 444
E.3 Tall Gaussian Matrices Are Well-Conditioned 445
E.4 Matrix Large Deviations 446

References 451

List of Symbols

$\Omega(n)$	"Big-Omega" means lower bounded by $C \cdot n$ for some constant C.
$\Theta(n)$	"Big-Theta" means lower bounded by $c \cdot n$ for some constant c and upper bounded by $C \cdot n$ for some constant $C > c$.
$O(n)$	"Big-O" means upper bounded by $C \cdot n$ for some constant C.
$o(n)$	"little-o" means ultimately smaller than n.
$\nabla f(\boldsymbol{x})$	The gradient of a differentiable function f at $\boldsymbol{x}$.
$\nabla^2 f(\boldsymbol{x})$	The Hessian of a twice-differentiable function f at $\boldsymbol{x}$.
$\partial f(\boldsymbol{x})$	Subdifferential of f at $\boldsymbol{x}$.
$\mathcal{A}, \mathcal{B}, \mathcal{P}$	General linear maps. These act on elements of their domain via square brackets, e.g., $\mathcal{A}[\boldsymbol{X}]$.
$\mathcal{P}_\Omega$	The projection operator of a matrix onto the coordinate subspace indexed by Ω.
$\mathcal{P}_S$	Orthonormal projector onto a subspace of a more general Hilbert space.
maximize $-(x+1)^2$	Unconstrained maximization.
minimize $(x+1)^2$	Unconstrained minimization.

$\text{minimize} \;\; \lambda \|\boldsymbol{x}\|_1 + \frac{1}{2}\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_2^2$ Lasso.

$$\begin{array}{ll} \text{minimize} & f(\boldsymbol{x}) \\ \text{subject to} & g(\boldsymbol{x}) \le 0,\ h(\boldsymbol{x}) = 0. \end{array}$$ Constrained minimization.

$$\begin{array}{ll} \text{minimize} & \|\boldsymbol{x}\|_1 \\ \text{subject to} & \boldsymbol{A}\boldsymbol{x} = \boldsymbol{y}. \end{array}$$ Basis pursuit.

$$\begin{array}{ll} \text{minimize} & \|\boldsymbol{x}\|_1 \\ \text{subject to} & \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|_2^2 \;\le\; \varepsilon^2. \end{array}$$ Basis pursuit denoising (BPDN).

$$\begin{array}{ll} \text{minimize} & \|\boldsymbol{X}\|_* \\ \text{subject to} & \mathcal{A}[\boldsymbol{X}] = \boldsymbol{y}. \end{array}$$ Affine rank minimization.

$$\begin{array}{ll} \text{minimize} & \|\boldsymbol{X}\|_* \\ \text{subject to} & \mathcal{P}_\Omega[\boldsymbol{X}] = \boldsymbol{Y}. \end{array}$$ Matrix completion.

$$\begin{array}{ll} \text{minimize} & \|\boldsymbol{X}\|_* + \lambda\|\boldsymbol{E}\|_1 \\ \text{subject to} & \boldsymbol{X} + \boldsymbol{E} = \boldsymbol{Y} \end{array}$$ Principal component pursuit (PCP).

$\boldsymbol{x}_0$, $\boldsymbol{X}_0$	Ground truth solutions.
$\boldsymbol{x}_k, \boldsymbol{x}_{k+1}$, $\boldsymbol{X}_k, \boldsymbol{X}_{k+1}$	Estimates at the k-th or the $(k+1)$-th iteration of an algorithm.
$\hat{\boldsymbol{x}}$, $\hat{\boldsymbol{X}}$	Estimated solutions (to an optimization problem).
$\boldsymbol{x}_\star$, $\boldsymbol{X}_\star$	Converged solutions of an iterative algorithm.
$\hat{\boldsymbol{x}} \in \arg\min \;\; f(\boldsymbol{x})$.	Set of minimizers of a function $f(\cdot)$.
$\boldsymbol{x}_\star = \arg\min \;\; f(\boldsymbol{x})$.	Shorthand when the minimizer of f is unique.
$\mathbb{E}\left[\cdot \mid \cdot\right]$	Conditional expectation
$\mathbb{P}\left[\, X > 1 \mid X < 2 \,\right] = 0$	Conditional probability
$\mathbb{E}\left[\cdot\right]$	Expectation
$\mathbb{1}_{x \le 3}$	Indicator for an event.
$\boldsymbol{e}$, $\boldsymbol{E}$	A gross error
$\boldsymbol{z}$, $\boldsymbol{Z}$	Noise
$\mathbb{P}\left[\, X > t \,\right] < \exp(-t^2/2)$, $\mathbb{P}_{\boldsymbol{A} \sim \text{i.i.d.} \mathcal{N}(0,1)}\left[\, \sigma_1(\boldsymbol{A}) > \sqrt{m} + \sqrt{n} + t \,\right] \le \exp(-t^2/2)$	Probability
$[k]$	The set $\{1, \ldots, k\}$
$\mathbb{C}$	The complex numbers
$\mathbb{R}$	The real numbers
$[\boldsymbol{x}]_k$	A best k-term approximation to $\boldsymbol{x}$.
$\boldsymbol{X}_{*,J}$	Shorthand for the column submatrix indexed by J.

$\mathbf{0}$	The zero vector or matrix, depending on context.
$\mathbf{1}$	The all ones vector or matrix, depending on context.
$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^*$	The singular value decomposition of $\boldsymbol{A}$. Prefer the "skinny" form. If $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ and $\operatorname{rank}(\boldsymbol{A}) = r$, $\boldsymbol{U} \in \mathbb{R}^{m\times r}$, $\boldsymbol{\Sigma} \in \mathbb{R}^{r\times r}$, and $\boldsymbol{V} \in \mathbb{R}^{n\times r}$.
$\boldsymbol{A} \succ \boldsymbol{B}$	Strict semidefinite order.
$\boldsymbol{A} \succeq \boldsymbol{B}$	The semidefinite order, i.e., $\boldsymbol{A} - \boldsymbol{B}$ is semidefinite.
$\boldsymbol{A}^\dagger$	The pseudoinverse of an arbitrary matrix $\boldsymbol{A}$.
$\boldsymbol{a}^*$, $\boldsymbol{A}^*$	The transpose of a vector $\boldsymbol{a}$ or a matrix $\boldsymbol{A}$.
$\boldsymbol{A}^{-1}$	The inverse of a nonsingular matrix $\boldsymbol{A}$.
$\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n$	The standard basis vectors for $\mathbb{R}^m$.
$\boldsymbol{I}$	The identity matrix.
$\boldsymbol{L}, \boldsymbol{S}$	$\boldsymbol{L}$ indicates a low-rank matrix, and $\boldsymbol{S}$ a sparse matrix.
$\boldsymbol{P} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^*$	The eigenvector decomposition of a symmetric matrix $\boldsymbol{P} \in \mathbb{R}^{m\times m}$. Here, $\boldsymbol{\Lambda}$ is diagonal, and $\boldsymbol{U} \in \mathbb{R}^{m\times m}$ with $\boldsymbol{U}^*\boldsymbol{U} = \boldsymbol{I}$.
$\boldsymbol{P}_I$	Abuse of notation for the projection (matrix) of a vector onto the coordinate subspace indexed by I.
$\boldsymbol{X}, \boldsymbol{Y}$	Matrices
$\boldsymbol{x}, \boldsymbol{y}$	Vectors
$\boldsymbol{X}_{i,j}$	The (i, j) element of matrix $\boldsymbol{X}$. Where possible, use i for the first index, j for the second index.
$\boldsymbol{\mathcal{X}}$	Tensors (of order > 2)
$\|\cdot\|_\diamond^*$	The dual norm of $\|\cdot\|_\diamond$.
$\|\boldsymbol{X}\|_*$	The nuclear norm.
$\|\boldsymbol{X}\|_{\ell^1\to\ell^2}^*$	The dual norm of the $\ell^1 \to \ell^2$ operator norm, $\sum_j \|\boldsymbol{X}\boldsymbol{e}_j\|_2$.
$\|\boldsymbol{X}\|_{\ell^1\to\ell^p}$	The $\ell^1 \to \ell^p$ operator norm, $\max_j \|\boldsymbol{X}\boldsymbol{e}_j\|_p$.
$\|\boldsymbol{X}\|_{\ell^2\to\ell^\infty}$	The $\ell^2 \to \ell^\infty$ operator norm, $\max_i \|\boldsymbol{e}_i^*\boldsymbol{X}\|_2$.
$\|\boldsymbol{X}\|_F$	The Frobenius norm.
$\|\boldsymbol{x}\|_p$	The vector ℓ^p norm
$\|\boldsymbol{X}\|$	The ℓ^2 operator norm, $\sigma_1(\boldsymbol{X})$.

$\|\mathcal{A}\|_{V\to W}$	The operator norm of $\mathcal{A}$, as an operator from normed space V to normed space W.
$\operatorname{null}(\boldsymbol{A})$	The null space of $\boldsymbol{A}$.
$\operatorname{range}(\boldsymbol{A})$	The range (column space) of $\boldsymbol{A}$.
$\operatorname{range}(\boldsymbol{A}^*)$	The row space of $\boldsymbol{A}$.
$\boldsymbol{X}_{I,*}$	Shorthand for the row submatrix indexed by I.
$\mathcal{S}_+^n$	The cone of symmetric positive semidefinite matrices of size $n \times n$.
$\operatorname{sign}(\boldsymbol{x})$	The signs of a vector $\boldsymbol{x} \in \mathbb{R}^n$, in $\{-1, 0, 1\}^n$.
$\boldsymbol{X}_{I,J}$	For $\boldsymbol{X} \in \mathbb{R}^{m\times n}$, the square submatrix index by $I \subseteq [m]$, $J \subseteq [n]$.
$\operatorname{supp}(\boldsymbol{x})$	For $\boldsymbol{x} \in \mathbb{R}^n$, the indices of the nonzero entries, $\subseteq [n]$.
a, b, c, x, y, A, B, C	Scalars
$C_1, C_2, \ldots$	Large constants.
$c_1, c_2, \ldots$	Small constants.