Chapter 13
Final Words

Regarding the fundamental investigations of mathematics, there is no final ending … no first beginning.

—Felix Klein

As we have stated from the very beginning of this book, the ultimate goal of our quest is to be able to effectively and efficiently extract low-dimensional structures in high-dimensional data. Our intention is for this book to serve as an introductory textbook for readers who are interested in modern data science and engineering, including both its mathematical and computational foundations as well as its applications. By using what is arguably the most basic and useful class of structures, i.e., linear subspaces, this book introduces some of the most fundamental geometrical, statistical, and optimization principles for data analysis. While these mathematical models and principles are classical and timeless, the problems and results presented in this book are rather modern and timely. Compared with classical methods for learning low-dimensional subspaces (such as PCA (Jolliffe 1986)), the methods discussed in this book significantly enrich our data analysis arsenal with modern methods that are robust to imperfect data (due to uncontrolled data acquisition processes) and can handle mixed heterogenous structures in the data.

In this final chapter, we discuss a few related topics that are not explicitly covered in this book because many of them are still open and active research areas. Nonetheless, we believe that these topics are all very crucial for the future development of modern data science and engineering, and the topics covered in this book serve as a good foundation for readers to venture into these more advanced topics.

R. Vidal et al., *Generalized Principal Component Analysis*, Interdisciplinary Applied Mathematics 40, DOI 10.1007/978-0-387-87811-9_13

13.1 Unbalanced and Multimodal Data

In practical applications, data are often highly imbalanced across classes. In the face clustering example we have used throughout the book, the number of face images from each individual may vary from individual to individual. As a result, when subspace clustering methods are applied to such imbalanced data, they may introduce a bias toward the class that has more samples and may have low performance on the minority class. Resampling methods—whereby underrepresented classes are oversampled and overrepresented classes are undersampled (He and Garcia 2009; He and Ma 2013)—can be applied to make the samples balanced. However, these methods can fail, because they completely ignore low-dimensional structures that are common in multiclass data. A promising approach to handling this issue is to automatically select a small subset of representatives for a large data set using sparse representation techniques (Elhamifar et al. 2012b,a). Interestingly, such methods are able to exploit multisubspace structures and guarantee that sufficiently many samples from each subspace are selected as representatives.

Another limitation of the techniques described in this book is that they require the data to come from the same modality. In practice, the desired information is often buried in various complementary types of data, say a combination of texts, audios, and images. How can we convert these different types of data into a common representation so that we can apply some of the methods in this book? So far, most of the techniques for handling multimodal data (say those popular in the multimedia literature) first extract features from each data type and then simply concatenate the features for analysis. It seems that there is increasing need to put information fusion from different data types on a firm theoretical and algorithmic foundation. Recent work in the area of domain adaptation aims to address this challenge (Patel et al. 2014; Jhuo et al. 2012; Qiu et al. 2012; Shekhar et al. 2013).

13.2 Unsupervised and Semisupervised Learning

According to the ontology of machine learning, all methods introduced in this book belong to the category of *unsupervised learning*. That is, they try to automatically learn the subspace structures of the data set without any manual labeling of the data classes[1] or manual setting of the model parameters. The reason for favoring unsupervised learning in the modern era of Big Data era is obvious: it is cost- and time-prohibitive to manually label massive data sets.

Nevertheless, in many practical situations and tasks, it is reasonable to assume that a small portion of the data set can be properly labeled in advance. Mathematically, the difficulty of the learning task can be dramatically alleviated even if a

[1]For instance, label whether a data point is an outlier; or label which subspace a data point belongs to in advance.

tiny subset of the data are labeled. We have seen some concrete examples in this book that support this view. In the algebraic subspace clustering method described in Chapter 5, we saw that although identifying individual subspaces through factorizing the vanishing polynomials (the vanishing ideal) is computationally intractable, the task can be significantly simplified once we are able to identify a sample point that belongs to one of the subspaces. Similar situations may naturally arise in many practical tasks. For instance, each Facebook user may have a few of the photos in his or her album labeled properly, yet it is desirable to use all the images in the Facebook repository (labeled or unlabeled) to build an effective face recognition or face labeling system.

Naturally, improving the effectiveness and efficiency of the subspace learning algorithms described in this book in the semisupervised learning setting will be a very meaningful and useful direction for future investigation. In particular, there is a need to develop principles that provide good guidelines for data sampling and labeling in such new settings, which, to the best of our knowledge, is still lacking.

13.3 Data Acquisition and Online Data Analysis

In this book, we have assumed that the data have all been collected in advance and have already been converted to a vector or matrix form ready for analysis. This may not be the case in many practical situations. For many demands of data analysis on the Internet or sensor networks, new data are accumulated on a daily basis and need to be stored, processed, and analyzed together with all the data that have been collected before. One natural example is how to analyze video streams from a network of cameras in a metropolitan area, either for traffic violations, security surveillance, or crime investigations. There is an obvious need for developing a real-time or online version of all the data analysis algorithms so that we can learn structures of the data adaptively as new data arrive and as the data structures evolve in time.

Toward the end of the book, in Chapters 11 and 12, we touched on applications of analyzing dynamical data such as videos and hybrid linear dynamical systems. However, to apply the methods in this book, we typically have to process such data in a *batch fashion*. To our knowledge, in the literature, there have already been good progress made toward developing online versions of some of the algorithms featured in this book, e.g., robust principal component analysis (Feng et al. 2013). There has also been good success in applying sparse representation and data clustering to real-time tasks such as object tracking in videos (Zhang et al. 2014). There has also been good progress on developing online versions of the algebraic subspace clustering algorithm for applications in online hybrid system identification (Vidal 2008). Nevertheless, how to develop online data analysis methods in a systematic and principled fashion remains an active research area.

One important issue associated with online data processing is how to control the data acquisition process so that we can more effectively collect the most informative

samples for the task at hand. If we could have some control over what data to collect and how to collect them, the subsequent data analysis tasks could potentially be dramatically simplified. This is one of the main messages advocated and supported by compressive sensing theory (Candès 2006; Baraniuk 2007).

13.4 Other Low-Dimensional Models

The class of models studied in this book, although very fundamental, can become inadequate for practical data sets that exhibit more sophisticated structures. As we have studied in Chapter 4, linear subspaces are no longer effective for data sets that have significant nonlinear structures. In such cases, the linear subspace model of PCA needs to be replaced with a low-dimensional surface or submanifold. However, although we have seen in Chapter 4 how such a nonlinear manifold can be learned through parametric or nonparametric techniques, we never dealt with data that may lie on a mixture of nonlinear manifolds.

Union of Manifolds
Note that a union of manifolds is a much more general (and expressive) class of models, which is also known in the literature as *stratifications* (Haro et al. 2008, 2006). Learning manifolds and stratifications remains an active research area, and many effective algorithms have been proposed so far. However, the theory and algorithms for manifold and stratification learning are still far from having reached the same level of maturity as those for subspace models covered in this book. Existing methods for clustering data in a union of manifolds include generalizations of the manifold learning algorithms discussed in Chapter 4, such as (Souvenir and Pless 2005), which is based on alternating minimization, and the locally linear manifold clustering (LLMC) algorithm (Polito and Perona 2002; Goh and Vidal 2007), which we discussed in Chapter 7 in the context of affine subspaces, but which generalizes to nonlinear manifolds. Another algorithm is sparse manifold clustering and embedding (SMCE) (Elhamifar and Vidal 2011), which generalizes the sparse subspace clustering algorithm discussed in Chapter 8. However, as stated before, a theoretical analysis of the conditions under which these methods give the correct clustering is still missing. Finally, there are also extensions of both LLMC and SSC to Riemannian manifolds, which have appeared in (Goh and Vidal 2008) and (Cetingül et al. 2014), respectively.

Compressive Sensing and Decomposable Structures
The rise of compressive sensing (Candès 2006; Baraniuk 2007) has brought to our attention a large family of low-dimensional structures in high-dimensional spaces, the so-called *decomposable structures* (Negahban et al. 2010; Candès and Recht 2011). In a sense, sparse signals, low-rank matrices (low-dimensional subspaces), and mixture of subspaces are all special cases of such structures, as we have seen in Chapters 3 and 8. All decomposable structures have similarly nice geometric and statistical properties as sparse signals and low-rank matrices:

they all can be recovered from nearly minimum samples via tractable means (say convex optimization). In addition, those structures can be arbitrarily combined (sum, union, and intersection) to generate an even broader family of low-dimensional structures. Nevertheless, beyond sparse and low-rank models, our understanding of and practice with structures in this broad family remains rather limited to this day. There is already evidence indicating that many such low-dimensional models and structures will play important roles in future data analysis.

Deep Learning and Deep Neural Networks
If the rise of compressive sensing is due to a series of mathematical breakthroughs, the revival of deep learning (Hinton et al. 2006) is largely attributed to some empirical successes of deep neural networks in classifying practical data such as speeches and images (Jarret et al. 2009). Since low-dimensional linear maps (such as the auto-encoders or the convolutional neural networks) are the key building blocks for each layer of a deep neural network, knowledge given in this book about low-dimensional linear models serves as a good foundation for thoroughly studying properties of hierarchical linear models such as deep neural networks and the treelike graphical models we used in Chapter 9. Recent theoretical advances in the analysis of deep neural networks have indicated strong connections of learning deep neural networks with dictionary learning (Spielman et al. 2012; Sun et al. 2015), sparse regularization (Arora et al. 2014), and matrix/tensor factorization (Haeffele et al. 2014; Haeffele and Vidal 2015). There are good reasons to believe that such advances will eventually lead to a rigorous and profound mathematical theory for deep networks and deep learning, similar to what has been established for sparse models in compressive sensing.

13.5 Computability and Scalability

According to the 2014 Big Data report from the White House, "*We are only in the very nascent stage of the so-called 'Internet of Things'.*" Our government, society, industry, and scientific community have been suddenly inundated with unprecedentedly massive data sets from the Internet (texts, audios, images, and videos, etc.) that contain important information about our daily lives and businesses. This has presented tremendous opportunities and challenges for the information technology industry and community, which require *correct mathematical algorithms and computing technologies to effectively and efficiently analyze those massive data sets and extract useful information from them.*

From Intractable to Tractable
While this book has taken only a few baby steps toward meeting the grand challenge of big data analysis, we have touched on a number of of promising and significant areas of progress in that direction. As we may recall, the problem of generalizing PCA to data with incomplete or corrupted entries or to data from multiple subspaces

is in general a highly combinatorial problem that is computationally intractable.[2] For instance, we saw in Chapter 5 and Appendix C that a precise characterization of the geometric structures of general subspace arrangements requires sophisticated (algebraic) geometric techniques whose computational complexity explodes as the dimension or the size of the data set increases. Now, there has been a very long history of research attempting to tackle instances of the GPCA problem with greedy, heuristic, brute force, or ad hoc algorithms. Although some of these algorithms have produced good results for many practical instances of the problem, one must realize that such algorithms do not provide any strong guarantee of success for general cases.[3] Because of this, at the beginning phase of our study of GPCA, we were wondering ourselves whether we would have to live with the fact that there will never be tractable algorithms for solving these GPCA problems with both correctness and efficiency guarantees. Fortunately, this did not turn out to be the case. With the help of more advanced statistical and computational tools from compressive sensing and convex optimization, researchers were able to develop tractable and efficient algorithms that provide provably correct solutions to the GPCA problems under broad conditions (see Chapters 3 and 8). Along the way, we have begun to realize how limited our understanding of high-dimensional data sets was and how surprisingly optimistic the situation has turned out to be.

From Tractable to Practical

However, having tractable solutions does not mean that the existing algorithms can already meet the modern challenge of big data analysis. Most of the algorithms introduced in this book are capable of handling data size or dimension up to the order of 10^4–10^5 on a typical computer. There has been tremendous effort in the computational community to speed and scale up core computational components heavily utilized by algorithms introduced in this book, including SVD for robust PCA or spectral clustering and ℓ_1 minimization for SSC. Many of the Internet-size data sets and problems require the scaling up of those algorithms by at least a few orders of magnitude. Hence, it is extremely important to investigate alternative optimization techniques that are more suitable for parallel and distributed computing and require less communication and memory. The drive for ever more scalable methods has become the source of inspiration for many ingenious new results in modern high-dimensional statistics and parallel optimization. For instance, the new factorization method mentioned in Chapter 3 has resulted from the effort to try to scale up the matrix completion or matrix recovery problem, instead of relying on the relatively expensive SVD. Recent promising generalizations of this approach have appeared in (Bach et al. 2008; Bach 2013; Haeffele et al. 2014; Udell et al. 2015). The search for ever more efficient and scalable sparse recovery algorithms has revolutionized optimization in the past few years with many new

[2]Strictly speaking, both problems are NP-hard in their general cases.

[3]One must be aware that success on instances can never be used as justification for the correctness of a proposed method.

parallel and distributed algorithms that are able to be implemented on commercial cloud computing platforms (Deng et al. 2013; Peng et al. 2013). Hence, we have sufficient reasons to be optimistic that for most methods and algorithms introduced in this book, researchers will be able to implement them and make them available to everyone on typical cloud computing platforms (such as the Hadoop MapReduce and the Spark systems) in the near future.

13.6 Theory, Algorithms, Systems, and Applications

As the demand for big data analysis is driven by many Internet-scale or world-scale applications, the ever more popular and powerful cloud computing platforms can be viewed as necessary technological infrastructures to support such tasks. However, big data and cloud computing would not have generated so much excitement in the scientific and research communities if they had required nothing more than scaling up what we used to do in the past. As this book has demonstrated, the challenges of analyzing massive high-dimensional data sets under uncontrolled engineering conditions has pushed researchers into the new realm of high-dimensional geometry, statistics, and optimization. We have begun to understand phenomena that were never imagined in classical low-dimensional settings or for tasks with small data sets.

The rise of compressive sensing and sparse representation has begun to provide researchers with a solid theoretical foundation for understanding the geometric and statistical properties of large high-dimensional data sets, whereas the revival of deep learning has begun to provide researchers with efficient computational platforms for handling practical (reinforced) learning tasks with large-scale, high-dimensional input data sets. Almost around the same time, the quest to seek tractable and efficient algorithms is revolutionizing optimization tools needed to learn such complex models and analyze such massive data sets. As we have mentioned before, many such optimization and learning algorithms can be easily implemented on modern cloud computing platforms, and hence can be scaled up to arbitrary sizes.

All these exciting developments make us believe that we are witnessing a *perfect storm* that takes place only occasionally in the history of science and engineering, whereby fundamental mathematical theories and significant engineering endeavors are fueling each other's explosive development. Never before have we seen long-isolated research fields in mathematics, statistics, optimization algorithms, computer systems, and industrial applications work so closely together on a common set of challenges. As a result, every field is making progress at an unprecedented rate, feeding on or fueling the progress and success of other fields. We anticipate that this trend will continue for quite some time, until a new body of scientific and engineering knowledge is fully developed. We hope that this book helps scientists and researchers move one step closer toward that grand goal.