

ALGORITHMS AND THEORY FOR CLUSTERING AND
NONCONVEX QUADRATIC PROGRAMMING

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF ELECTRICAL
ENGINEERING
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Mahdi Soltanolkotabi
August 2014

Abstract

In this dissertation we discuss three problems characterized by hidden structure or information. The first part of this thesis focuses on extracting subspace structures from data. Subspace Clustering is the problem of finding a multi-subspace representation that best fits a collection of points taken from a high-dimensional space. As with most clustering problems, popular techniques for subspace clustering are often difficult to analyze theoretically as they are often non-convex in nature. Theoretical analysis of these algorithms becomes even more challenging in the presence of noise and missing data. We introduce a collection of subspace clustering algorithms, which are tractable and provably robust to various forms of data imperfections. We further illustrate our methods with numerical experiments on a wide variety of data segmentation problems.

In the second part of the thesis, we consider the problem of recovering the seemingly hidden phase of an object from intensity-only measurements, a problem which naturally appears in X-ray crystallography and related disciplines. We formulate the problem as a non-convex quadratic program whose global optimum recovers the phase information exactly from a near minimal number of magnitude-only measurements. To solve this non-convex problem, we develop an iterative algorithm that starts with a careful initialization and then refines this initial estimate by iteratively applying novel update rules. The main contribution is that we show that the sequence of successive iterates provably converges to the global optimum at a geometric rate so that the proposed scheme is efficient both in terms of computational and data resources. We also show that this approach is stable vis a vis noise. In theory, a variation on this scheme leads to a near-linear time algorithm for a physically realizable model based on coded

diffraction patterns. In this part of the thesis we also prove similar results about two other approaches, the first one is based on convex optimization and the second one is inspired by the error reduction algorithm of Gerchberg-Saxton and Fienup. We illustrate the effectiveness of our methods with various experiments on image data. Underlying the analysis of this part of the thesis are insights for the analysis of non-convex optimization schemes that may have implications for computational problems beyond phase retrieval.

In the third part of the thesis, we look at two related problems involving coherent and redundant dictionaries. The first problem, is about the recovery of signals from under-sampled data in the common situation where such signals are not sparse in an orthonormal basis, but in a coherent and redundant dictionary. We focus on a formulation of the problem where one minimizes the ℓ_1 norm of the coefficients of the representation of the signal in the dictionary subject to the measurement constraints, a.k.a. the synthesis problem. For this formulation we characterize the required number of random measurements in terms of geometric quantities related to the dictionary. Furthermore, we connect this problem to the *denoising* problem where instead of under-sampled measurements of the signal we observe a noisy version of it. In this case we characterize the reconstruction error obtained by using the over-complete dictionary for denoising and show that it depends on the same geometric quantities that affect the number of measurements in the synthesis problem. The second problem concerns sparse recovery with coherent and redundant dictionaries which appears in a variety of applications such as microscopy, astronomy, tomography, computer vision, radar, and seismology. Our results show that sparse recovery via ℓ_1 minimization is effective in these dictionaries even though these dictionaries have maximum pair-wise column coherence very close to 1, i.e. they contain almost identical columns. This holds with the proviso that the sparse coefficients are not too clustered. This general theory, when applied to the special case of low pass Fourier (a.k.a. super-resolution), allows for less restrictive requirements when compared with recent literature with significantly shorter proofs.

Acknowledgements

I am very grateful to my advisor Emmanuel Candès for piquing my interest in applied math and statistics through his papers as an undergrad and later on throughout my Ph.D. at Stanford. Emmanuel's insistence on tackling difficult yet relevant problems was ideal for me. His advice and feedback on research and scientific writing—even when I disagreed with them—were invaluable as it would often force me into the thinking position. I am also thankful to him for giving me the freedom to pursue eclectic and sometimes unorthodox research directions which helped me find an independent voice in research. Finally, Emmanuel's dedication to making theoretical research accessible to and understandable by a wide audience through meticulous editing of papers, presentations and class lectures has been nothing short of exemplary for me.

I would like to express my gratitude to Stephen Boyd, Trevor Hastie, Andrea Montanari, and David Tse for serving on my thesis committee, as well as many other professors whom I took classes with at Stanford. I would like to thank Andrea Montanari for many enthusiastic presentations, lectures and classes at Stanford. His depth and diversity of knowledge made his research presentations/classes a very enriching and rewarding experience. I am grateful to Trevor Hastie for teaching me statistical learning through his classes and his fantastic book as well as his support during the past year. I would also like to thank Stephen Boyd for giving me the opportunity to teach his class during my second year as a grad student which helped me get a better grasp of convex optimization early on.

I have been fortunate to have had great peers throughout my education. Countless peers and friends during undergrad at Sharif EE who with their uniform excellence kept me motivated and focused on my studies. Special thanks goes to Saber Saleh

Kaleybar and Adel Javanmard my collaborators through many course assignments and projects. I would also like to thank several fellow graduate students as well as my group members past and present at Stanford: Carlos Fernandez-Granda, Xiaodong Li, Adel Javanmard, Yaniv Plan, Mark Davenport, Ewout Van Den Berg, Stephen Becker, Rina Foygel, Lester Mackey, Veniamin Morgenshtern, Deanna Needell, Reinhard Heckel, Carlos Sing-Long, Vlad Voroninski, Alexandra Chouldechova, Weijie Su, Kahye Song, Yuxin Chen, and Kenji Nakahira.

Last but not least, I am eternally grateful to my parents and my sister, for their endless love and support. Without you, I would not be here today.

Contents

Abstract	iv
Acknowledgements	vi
I Subspace Clustering	1
1 The subspace clustering problem	2
1.1 Problem formulation	4
2 Applications in computer vision	7
2.1 Motion segmentation	7
2.2 Face clustering	10
2.3 Temporal segmentation of motion capture data	10
3 Prior art in subspace clustering	12
3.1 Generalized PCA (GPCA): an algebraic approach	12
3.1.1 Representing union of subspaces via homogeneous polynomials	13
3.1.2 Fitting polynomials to the data samples	14
3.1.3 Obtaining the subspace bases by differentiation	16
3.1.4 Choosing one point per subspace by polynomial division	17
3.1.5 The GPCA algorithm	19
3.1.6 Some theory for GPCA	19
3.1.9 Pros and cons of GPCA	21
3.2 Iterative algorithms	22

3.2.1	The K-subspaces algorithm	22
3.2.2	Pros and cons of K-subspaces	23
3.3	Statistical algorithms	24
3.3.1	Mixture of probabilistic PCA	24
3.3.2	Agglomerative lossy compression	25
3.3.3	Random sampling consensus	27
3.4	Spectral clustering-based algorithms	28
3.4.1	Cosine-based affinity	31
3.4.2	Factorization-based affinity	32
3.4.3	Local subspace affinity and spectral local best-fit flats	32
3.4.4	Spectral curvature clustering	34
4	Robust subspace clustering	36
4.1	The SSC scheme	36
4.2	From SSC to Robust Subspace Clustering	37
4.2.1	Affine subspace clustering	38
4.3	Performance metrics for similarity measures	39
4.4	Noisy data	39
4.4.1	LASSO with data-driven regularization	40
4.4.2	The Bias-corrected Dantzig Selector	47
4.5	Gross outliers	52
4.6	Missing data	53
4.6.1	Detailed implementation	55
4.7	Sparse corruption	56
5	Theory	57
5.1	Modeling assumptions	57
5.1.1	Models for the clean data points	58
5.1.2	Models for corruption	58
5.2	What makes clustering hard?	59
5.2.1	Distance/affinity between subspaces	59
5.2.2	Distribution of points on each subspace and sampling density .	61

5.3	Noiseless data	63
5.3.1	Deterministic model	64
5.3.2	Semi-random model	67
5.3.3	Fully-random model	68
5.3.4	Comparison with previous results on SSC	70
5.4	Segmentation in the presence of noise	72
5.4.1	Main results	73
5.5	Segmentation with gross outliers	76
5.5.1	Comparison with other theoretical results	79
5.6	Towards segmentation with missing data	80
5.6.1	When is subspace clustering with missing data possible?	80
5.6.2	What is the correct choice of λ ?	82
5.6.3	Guarantees for subspace clustering with missing data	83
5.7	Comparison with other schemes	84
6	Numerical experiments	87
6.1	Error metrics	87
6.2	Synthetic experiments	88
6.2.1	Segmentation with noiseless data	89
6.2.2	Segmentation with outliers	93
6.2.3	Segmentation with missing data	94
6.3	Experiments on temporal segmentation of motion capture data	97
6.4	Experiments on motion segmentation data	102
6.5	Experiments on face clustering	103
6.6	Experiments on cancer data	103
6.6.1	No missing entries	104
6.6.2	With missing entries	106
6.7	Experiments on Flickr photos of animals	107
7	Proofs	109
7.1	Linear programming theory	109
7.2	Proofs for noiseless data	111

7.2.1	Proof of Theorem 5.3.5	112
7.2.2	Proof of Theorem 5.3.6	115
7.2.3	Proof of Theorem 5.3.7	119
7.3	Proof of results with noise	120
7.3.1	Intermediate results	121
7.3.2	Proof of Theorem 5.6.3	129
7.3.3	The size of the solution to the projected problem	132
7.3.4	Proof of Theorem 5.6.4	133
7.4	Proof of results with gross outliers	136
7.4.1	Background on Geometric Functional Analysis	136
7.4.2	Proof of Theorem 5.5.2	138
7.4.3	Proof of Theorem 5.5.1	142
7.5	Proof of results with missing data	143
7.5.1	Proof of Theorem 5.6.2	143
II	Phase Retrieval	148
8	The generalized phase retrieval problem	149
9	Applications of phase retrieval	153
9.1	Applications in optical imaging	153
9.1.1	Some history	154
9.1.2	Coherent Diffraction Imaging (CDI)	154
9.2	Speckle imaging in astronomy	156
9.3	Blind channel estimation	158
10	Prior art in phase retrieval	160
10.1	When is the phase retrieval problem unique?	160
10.1.1	Fourier measurements	160
10.1.2	General measurements	162
10.2	Classical approaches to phase retrieval	163
10.2.1	Error reduction algorithm	163

10.2.2 Solvent flipping algorithm	165
10.2.3 Hybrid input-output algorithm	165
11 Phase retrieval via convex relaxation	167
11.1 Convex relaxation	167
11.2 Coded diffraction patterns	168
11.3 Modeling assumptions	170
11.4 Main results	171
11.4.1 Noiseless measurements	171
11.4.2 Noisy measurements	172
11.5 Comparison with previous work	173
11.5.1 Comparison with prior art using convex relaxation	173
11.5.2 Other approaches to phase retrieval and related works	174
11.6 Discussion	175
12 Phase retrieval via Wirtinger Flow	177
12.1 Algorithm: Wirtinger Flow	177
12.1.1 Minimization of a non-convex objective	178
12.1.2 Initialization via a spectral method	179
12.1.3 Wirtinger flow as a stochastic gradient scheme	180
12.2 Exact phase retrieval via Wirtinger flow	182
12.2.1 Theory for the Gaussian model	183
12.2.2 Theory for the Coded Diffraction Model	184
12.3 Stable phase retrieval via Wirtinger flow	186
12.4 Comparison with other non-convex schemes	189
13 Error reduction via non-convex optimization	192
13.1 ER algorithm as non-convex optimization	193
13.2 Some theory for the convergence of the ER algorithm	194
14 Numerical experiments	196
14.1 Models	196

14.1.1	Signal models	196
14.1.2	Measurement models	197
14.2	Synthetic experiments	197
14.2.1	Phase transition of Phase Lift using CDP measurements	198
14.2.2	Phase transition of Wirtinger Flow using Gaussian and CDP measurements	199
14.2.3	Noisy measurements	200
14.3	Performance on natural images	202
14.4	3D molecules	205
15	Proofs	209
15.1	Proofs for PhaseLift with CDP measurements	209
15.1.1	Preliminaries	210
15.1.2	Certificates	211
15.1.3	Robust injectivity	213
15.1.4	Dual certificate construction via the golfing scheme	218
15.1.5	Proof of Lemma 15.1.8	220
15.2	Proof of stability of PhaseLift with CDP measurements	222
15.3	Proofs for Wirtinger flow	225
15.3.1	Preliminaries	225
15.3.2	Formulas for the complex gradient and Hessian	227
15.3.3	Expectation and concentration	228
15.3.4	General convergence analysis	230
15.3.5	Proof of the regularity condition	232
15.3.6	Proof of the local curvature condition	233
15.3.7	Proof of the local smoothness condition	238
15.3.8	Wirtinger flow initialization	241
15.3.9	Initialization via resampled Wirtinger Flow	242
15.4	Proofs of stability of Wirtinger flow	244
15.4.1	Proof of stability of the global optimum of the WF objective (Proof of Theorem 12.3.1)	244

15.4.2 Proof of stability of the WF initialization (Proof of first part of Theorem 12.3.2)	245
15.4.3 Proof of stability of the WF iteration updates (Proof of second part of Theorem 12.3.2)	247
15.5 Proofs for the error reduction algorithm	255
15.5.1 convergence of the iterations of the error reduction	257
15.5.2 Proof of stability of the global optimum of the WF objective (Proof of Theorem 12.3.1)	257
III Compressed Sensing, Denoising and Sparse Recovery with Coherent and Redundant Dictionaries	258
16 Background	259
17 Compressive sensing and denoising	261
17.1 Problem statement and preliminaries	261
17.1.1 Compressed sensing with coherent and redundant dictionaries	261
17.1.2 Denosing with coherent and redundant dictionaries	262
17.2 Theory	262
17.2.1 Some geometric definitions and assumptions	263
17.2.2 Theory for compressed sensing with coherent and redundant dictionaries	265
17.2.3 Theory for denoising with coherent and redundant dictionaries	266
17.2.4 Comparison with some related literature	267
18 Sparse recovery with coherent dictionaries	269
18.1 Two models and their connection	269
18.1.1 Continuous frequency model	269
18.1.2 Discrete frequency model	270
18.1.3 The connection	271
18.2 Continuous Super-resolution via TV-minimization	271

19 Proofs	274
19.1 Proof of compressive sensing with coherent dictionaries (Theorem 17.2.4)	274
19.1.1 Proof of the geometric null space property (Lemma 19.1.2) . . .	276
19.1.2 Interpretation of the geometric null space property	278
19.2 Proof of Lemma 17.2.5	280
19.3 Proof of denoising with coherent dictionaries (Theorem 17.2.6)	281
19.4 Proof of sparse recovery with highly coherent dictionaries	283
19.4.1 Proof of the connection (Theorem 18.1.5)	283
19.4.2 Proof of the continuous super-resolution problem (Theorems 18.2.1 and 18.2.2)	284
19.4.3 Proof of main lemmas	291
A Geometric Perspective on the subspace detection property	305
B Standard inequalities in probability	309
C Geometric Lemmas	311
D Sharpening Lemma 7.3.3 Asymptotically	313
E Proof of auxilary lemmas for establishing the exactness of PhaseLift with CDP measurements	317
E.1 Proof of Lemma 15.1.1	317
E.2 Proof of Lemma 15.1.2	319
E.3 Proof of Lemma 15.1.3	320
E.4 Proof of Lemma 15.1.4	320
F Extensions of proofs of PhaseLift to higher dimensions by tensoriza- tion	321
G Wirtinger derivatives	324
H Expectations and deviations	327
H.1 Proof of Lemma 15.3.1	327

H.2 Proof of Lemma 15.3.2	328
H.3 Proof of Lemma 15.3.3	328
H.4 Proof of Lemma 15.3.4	329
H.5 Proof of Corollary 15.3.5	333
H.6 Proof of Corollary 15.3.6	333
H.7 Proof of Lemma 15.3.7	334
H.8 Proof of Lemma 15.3.8	335
I The Power Method	336
Bibliography	338

List of Tables

6.1	Minimum clustering error.	99
6.2	Optimal parameters.	99
6.3	Clustering error (%) of different algorithms on the Hopkins 155 dataset.	103
6.4	Clustering error (%) of different algorithms on the Extended Yale B dataset.	103
6.5	Summary of the data.	104
6.6	Minimum clustering error for various algorithms and data sets.	105

List of Figures

1.1	Collection of points near a union of multiple subspaces.	3
2.1	Point trajectories of two moving cars and a nonmoving background across 4 different frames. Each color corresponds to a different object (cluster).	8
2.2	Left: eight activities performed by subject 86 in the CMU motion capture dataset: walking, squatting, punching, standing, running, jumping, arms-up, and drinking. Right: singular values of the data from three activities (walking, jumping, drinking) show that the data from each activity lie approximately in a low-dimensional subspace.	11
4.1	Average number of true discoveries normalized by subspace dimension for values of λ in an interval including the heuristic $\lambda_o = 1/\sqrt{d}$. (a) $\sigma = 0.25$. (b) $\sigma = 0.5$	44
4.2	Performance of LASSO for values of λ in an interval including the heuristic $\lambda_o = 1/\sqrt{d}$. (a) Average number of false discoveries normalized by $(n - d)$ (FPR) on all m sampled data points. (b) FPR for different subspace dimensions. Each curve represents the average FPR over those samples originating from subspaces of the same dimension. (c) Average number of true discoveries per dimension for various dimensions (TPR). (d) TPR vs. FPR (ROC curve). The point corresponding to $\lambda = \lambda_o$ is marked as a red dot.	45

4.3	Optimal values of (4.4.4) for 600 samples using $\tau = 2\sigma$. The first 100 values correspond to points originating from subspaces of dimension $d = 200$, the next 100 from those of dimension $d = 150$, and so on through $d \in \{100, 50, 20, 10\}$. (a) Value of $\ \beta^*\ _{\ell_1}$. (b) Value of $\ \beta^*\ _{\ell_1}/\sqrt{d}$.	47
4.4	Performance of the two-step procedure using $\tau = 2\sigma$ and $f(t) = \alpha_0 t^{-1}$ for values of α_0 around the heuristic $\alpha_0 = 0.25$. (a) False positive rate (FPR). (b) FPR for various subspace dimensions. (c) True positive rate (TPR). (d) TPR vs. FPR.	48
4.5	Performance of the bias-corrected Dantzig selector for values of λ that are multiples of the heuristic $\lambda_o = \sqrt{2/n} \sigma \sqrt{1 + \sigma^2}$. (a) False positive rate (FPR). (b) FPR for different subspace dimensions. (c) True positive rate (TPR). (d) TPR vs. FPR.	51
5.1	This picture shows how the distance between the subspaces affects the subspace clustering problem. Parts (a) and (c) depict a cloud of points. Parts (b) and (d) show the same points along with the subspaces they originate from. One can see visually that inferring the subspaces from the points is easier for parts (a) and (b) when compared with parts (c) and (d). This suggests that subspace clustering is more difficult when the subspaces are more aligned with each other.	60
5.2	This picture shows how the distribution of the points on each of the subspaces affects the subspace clustering problem. Parts (a) and (c) depict a cloud of points. Parts (b), (d) and (e) show possible subspace fits. One can see visually that inferring the subspaces from the points is easier for parts (a) and (b). Inferring the subspaces in parts (c)-(e) is significantly more challenges as it is not clear which of the two alternatives (part (d) vs. part(e)) is the correct choice. This suggests that subspace clustering is more difficult when the points on the subspaces align along lower dimensional subspaces.	62
5.3	Geometric representation of a dual point, see Definition 5.3.1.	64

5.4	Geometric representation of a dual direction. The dual direction is the dual point embedded in the ambient n -dimensional space.	65
5.5	Skewed distribution of points on a single subspace and ℓ_1 synthesis. .	66
5.6	Histograms of the true discovery values from the two step procedure with $\alpha_0 = 0.25$ (multiplied by \sqrt{d}). (a) $d = 200$. (b) $d = 20$	75
5.7	Plot of the threshold function (5.5.1).	77
6.1	Error metrics as a function of the dimension of the intersection.	90
6.2	Performance of the SSC algorithm for different values of the affinity and density of points per subspace. In all three figures, the horizontal axis is the density ρ , and the vertical axis is the maximum affinity $\max_{i \neq j} \text{aff}(S_i, S_j)$	92
6.3	Gaps in the eigenvalues of the normalized Laplacian as a function of subspace dimension.	93
6.4	Gap in the optimal values with $L = 2n/d$ subspaces. (a) $d = 5$, $n = 50$, $L = 20$. (b) $d = 5$, $n = 100$, $L = 40$. (c) $d = 5$, $n = 200$, $L = 80$	95
6.5	The fraction of correctly completed columns (with a tolerance of 10^{-5}), versus the fraction of missing entries δ for the bias-corrected Dantzig Selector and the algorithm suggested in [100].	96
6.6	Minimum clustering error (%) for each K in the baseline algorithm. .	98
6.7	Clustering error (%) for different values of λ and σ on trial 5 using RSC-N (a) 3D plot (minimum clustering error appears in red). (b) 2D cross sections.	100
6.8	Clustering error (%) for different values of λ and σ on trial 5 using RSC-D. (a) 3D plot (minimum clustering error appears in red). (b) 2D cross sections.	101
6.9	Box plot of the affinities between subspaces for trials 2 and 5.	102
6.10	Heat map plot of the gene expression level of the different groups of patients in the St. Jude Leukemia data set.	105
6.11	Clustering error (%) for different cancer data examples and different fractions of missing entries δ	106

6.12	Sample Flicker images from NUS-WIDE database.	107
6.13	Clustering errors on images of pairs of animals in NUS-WIDE dataset.	108
7.1	Illustration of Definitions 7.1.1 and 7.1.2. (a) Norm with respect to a polytope \mathcal{K} . (b) Polytope \mathcal{K} and its polar \mathcal{K}° .	110
9.1	An illustrative setup for Coherent Diffraction Imaging: A coherent wave diffracts from a sample, and produces a far-field diffraction pattern which corresponds to the magnitude of the Fourier transform of the sample.	155
9.2	An example of phase retrieval for speckle imaging. In (A) we see 10 sample speckle images of a double star (ϵ Lyrae). In (B) we see the high resolution image of the same star obtained through phase retrieval techniques. This figure is from [130, 203].	158
11.1	An illustrative setup for acquiring coded diffraction patterns.	169
12.1	Learning parameter μ_τ from (12.1.5) as a function of the iteration count τ ; here, $\tau_0 \approx 330$ and $\mu_{\max} = 0.4$.	182
14.1	Empirical probability of success based on 50 random trials for different signal/measurement models and a varied number of measurements. A value of L on the x-axis means that we have a total of $m = Ln$ samples.	198
14.2	Empirical probability of success based on 100 random trials for different signal/measurement models and a varied number of measurements. The coded diffraction model uses octanary patterns; the number of patterns $L = m/n$ only takes on integral values.	199
14.3	SNR versus relative MSE on a dB-scale for different kinds of signal/measurement models and algorithms. The linear relationship between SNR and MSE (on the dB scale) is apparent. The MSE behaves as in a well-conditioned least-squares problem.	202

14.4 Performance of the WF algorithm on three scenic images. Image size, computational time in seconds and in units of FFTs are reported, as well as the relative error after 300 WF iterations.	204
14.5 An illustrative setup of diffraction patterns.	205
14.6 Schematic representation and electron density map of the Caffeine molecule.	206
14.7 Schematic representation and electron density map of the Nicotine molecule.	207
14.8 Electron density $\rho(x_1, x_2, x_3)$ of the Caffeine molecule along with its projection onto the x_1x_2 -plane.	207
14.9 Reconstruction sequence of the projection of the Caffeine and Nicotine molecules along different directions. To see the videos please visit the author's website.	208
18.1 Plot of minimum possible OSR ($\eta(\Omega)$) under which TV minimization yields exact recovery as a function of the highest frequency Ω	273
19.1 Edge in red denotes face of polytope. Line in blue denotes null space of \mathbf{A} . Null space intersects with the interior of the polytope (synthesis fails).	278
19.2 2-D plane representing the row space of \mathbf{A} , projection of the face in red onto the row space of \mathbf{A} , and projection of the rest of the vertices of the polytope onto this subspace in blue.	278
19.3 Edge in red denotes face of polytope. Line in blue denotes null space of \mathbf{A} . Null space is tangent to the polytope (synthesis succeeds).	279
19.4 2-D plane representing the row space of \mathbf{A} , projection of the face in red onto the row space of \mathbf{A} , and projection of the rest of the vertices of the polytope onto this subspace in blue.	279
A.1 Illustration of ℓ_1 minimization when the subspace detection property holds. Same object seen from different angles.	306

A.2	Illustration of ℓ_1 minimization when the subspace detection property fails. Same object seen from different angles.	306
A.3	Geometric view of (5.3.1). The right figure is seen from a direction orthogonal to S_1	307
D.1	Left-hand side (blue) and right-hand side (red) of (D.0.3). The two curves intersect at $(\alpha_*, \delta_*) = (0.9254, 0.35476)$	316
F.1	An example of constructing an admissible two dimensional make of size 5×8 of the form $\mathbf{d1}^*$ and $\mathbf{1b}^*$ using one dimensional admissible masks of size 5 and 8.	323

Part I

Subspace Clustering

Chapter 1

The subspace clustering problem

Principal Component Analysis is a ubiquitous tool in modern data analysis with many applications in a variety of diverse fields ranging from neuroscience to computer vision. The reason for the widespread use of PCA is that it is a simple method for extracting relevant information from high-dimensional data sets. PCA is a method for reducing a complex data set to a lower dimension with the goal of revealing the sometimes hidden, simplified structures that often underlie it.

PCA finds a low-dimensional subspace which best fits a collection of points taken from a high-dimensional space. Such a procedure makes perfect sense as long as the data points are distributed around a *single* lower-dimensional subspace. In practice, however, the data points could be distributed around *multiple* low-dimensional subspaces as shown in Figure 1.1. For instance, a video could contain several moving objects, and different subspaces might be needed to model the motion of the different objects in the scene. Furthermore, the data points are unlabeled in the sense that we do not know in advance to which subspaces they belong to. Therefore, we need to simultaneously cluster these data into multiple subspaces and find a low-dimensional subspace approximating all the points in a cluster. This problem is known as *subspace clustering*. It can also be seen as a nonstandard clustering problem in which neighbors are not close according to a pre-defined notion of metric but rather belong to the same lower dimensional structure.

In recent years, numerous algorithms have been developed for subspace clustering

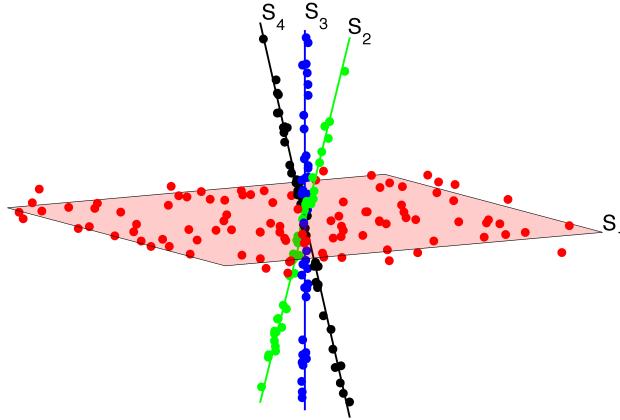


Figure 1.1: Collection of points near a union of multiple subspaces.

and applied to various problems in computer vision/machine learning [233] and data mining [198]. Subspace clustering techniques are used in fields as diverse as identification and classification of diseases [166], network topology inference [100], security and privacy in recommender systems [250], system identification [16], hyper-spectral imaging [75], hand-written digit recognition [124, 125], identification of switched linear systems [159, 196], and music analysis [144] to name just a few. We will review some of these methods along with their advantages/disadvantages in Section 3.

In addition to the many applications mentioned above, union of multiple subspaces (or *subspace arrangements* as they are known in math), and their topological complements, are very important classes of objects that have been studied in mathematics for centuries. Various aspects of the properties of subspace arrangements continue to be investigated and exploited in many mathematical fields such as algebraic geometry, algebraic topology, combinatorics and complexity theory, and graph and lattice theory [42, 43, 193].

Despite a growing number of approaches and experiments for subspace clustering problems in the past two decades, tractable subspace clustering algorithms either lack a theoretical justification, or are guaranteed to work under restrictive conditions

rarely met in practice. Furthermore, proposed algorithms are not always computationally tractable. Thus, an important issue is whether tractable algorithms that can (provably) work in less than ideal situations—that is, under severe corruptions (noise, missing data, and outliers) and relatively few samples per subspace—exist.

In this part we introduce a collection of subspace clustering algorithms, which are tractable and provably robust to various forms of data imperfections such as noise, missing data, and outliers. Furthermore, we demonstrate via numerical experiments that these algorithms, which are almost parameter free, are effective on a wide variety of data segmentation problems. The results presented in this part of the dissertation are based on three papers [59, 217, 218] which collectively provide the basis for the first provably robust and tractable algorithm for subspace clustering. We note that the text of Chapters 4, 5, 6, and 7 heavily borrows from these papers.

1.1 Problem formulation

We assume we are given data points in \mathbb{R}^n lying near a union of unknown subspaces; there are L affine subspaces S_1, S_2, \dots, S_L of dimension d_1, d_2, \dots, d_L centered at $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \dots, \boldsymbol{\mu}_L \in \mathbb{R}^n$. Stated differently, subspaces can be described as

$$S_\ell = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{x} = \boldsymbol{\mu}_\ell + \mathbf{U}^{(\ell)} \mathbf{c}\}, \quad \ell = 1, 2, \dots, L.$$

Here, $\boldsymbol{\mu}_\ell \in \mathbb{R}^n$ is an arbitrary point in subspace S_ℓ , $\mathbf{U}^{(\ell)} \in \mathbb{R}^{n \times d_\ell}$ is an orthonormal basis for subspace S_ℓ , and $\mathbf{c} \in \mathbb{R}^{d_\ell}$ is a low-dimensional representation for point \mathbf{x} . These subspaces, together with their dimensions, offsets and number are completely unknown to us. We are given a point set $\mathcal{Y} \subset \mathbb{R}^n$ consisting of N points $\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N \in \mathbb{R}^n$, which may be partitioned as

$$\mathcal{Y} = \mathcal{Y}_0 \cup \mathcal{Y}_1 \cup \dots \cup \mathcal{Y}_L, \tag{1.1.1}$$

for each $\ell \geq 1$, \mathcal{Y}_ℓ is a collection of N_ℓ vectors that are “close” to subspace S_ℓ . The careful reader will notice that we have an extra subset \mathcal{Y}_0 in (1.1.1) accounting for possible outliers. We will often gather the data points as columns of a matrix $\mathbf{Y} \in$

$\mathbb{R}^{n \times N}$. Given the point set \mathcal{Y} the goal is to

- (1) identify all of the outliers if they exist,
- (2) segment or assign each data point to a cluster, and
- (3) find the number of subspaces L , their dimensions d_1, d_2, \dots, d_L , the subspace bases $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \dots, \mathbf{U}^{(L)}$, and their offsets $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \dots, \boldsymbol{\mu}_L$.

Our model assumes that each point $\mathbf{y} \in \mathcal{Y}_1 \cup \mathcal{Y}_2 \cup \dots \cup \mathcal{Y}_L$ is of the form

$$\mathbf{y} = \mathbf{x} + \mathbf{z}, \quad (1.1.2)$$

where \mathbf{x} denotes the “clean” data point which belongs to one of the subspaces and \mathbf{z} denotes the corruption on this point. With obvious notation $\mathbf{Y} = \mathbf{X} + \mathbf{Z}$. We will consider five different forms of corruptions which appear naturally in many applications:

- *Noise-free data.* In this model all the data points reside exactly on one of the subspaces without any additional corruption. This corresponds to $\mathbf{z} = \mathbf{0}$ in (5.1.1).
- *Noisy data.* In this model each of the data points is close to (but not exactly on) one of the subspaces. This is equivalent to the model in (5.1.1) with $\|\mathbf{z}\|_{\ell_2}$ having a small value.
- *Missing data.* In this model some of the entries of each of the data points may be missing. Let $\Omega \subset \{1, 2, \dots, n\}$ denote the index of the entries that are revealed. Placing zeros for the missing entries, this form of corruption can also be modeled in the form of (5.1.1) where the vector \mathbf{z} takes the value zero for the revealed entries ($\mathbf{z}_{\Omega} = \mathbf{0}$) and nulls the value of the missing entries ($\mathbf{z}_{\Omega^c} = -\mathbf{x}_{\Omega^c}$), so that $\mathbf{y}_{\Omega} = \mathbf{x}_{\Omega}$ and $\mathbf{y}_{\Omega^c} = \mathbf{0}$.
- *Sparse corruptions.* In this model a few, unknown entries of each data point is grossly corrupted. This is equivalent to the model in (5.1.1) with \mathbf{z} a sparse vector with the non-zero entries representing the sparse corruptions.

- *Gross outliers.* In certain applications due to sensor failure or related reasons certain data points may be completely corrupted. As mentioned previously we model these outliers by the extra outlier set \mathcal{Y}_0 .

Chapter 2

Applications in computer vision

In this chapter we explain how subspace clustering arises naturally in three different applications in computer vision: motion segmentation, face clustering, and temporal segmentation of motion capture data.

2.1 Motion segmentation

A fundamental problem at the heart of many computer vision tasks is to infer 3D structure and movements of objects from a video sequence. Such video sequences often contain multiple moving objects in addition to camera motion. Thus, an important step in understanding a scene is to separate the video sequence into multiple regions corresponding to different moving objects known as *motion segmentation*. Standard computer vision techniques now allow us to track a set of feature points through a sequence of video frames. Therefore, the motion segmentation problem reduces to clustering the trajectories of those points according to different motions. The result of such a clustering across four frames is depicted in Figure 2.1.

Suppose we are given the trajectories of N_ℓ tracked feature points of a rigid object $\{(x_{fk}, y_{fk})\}_{f=1,2,\dots,F}^{k=1,2,\dots,N_\ell}$ from F 2D frames taken by a moving camera. We arrange these data points in a matrix $\mathbf{X}^{(\ell)} \in \mathbb{R}^{n \times N_\ell}$. Each column in $\mathbf{X}^{(\ell)}$ correspond to a feature

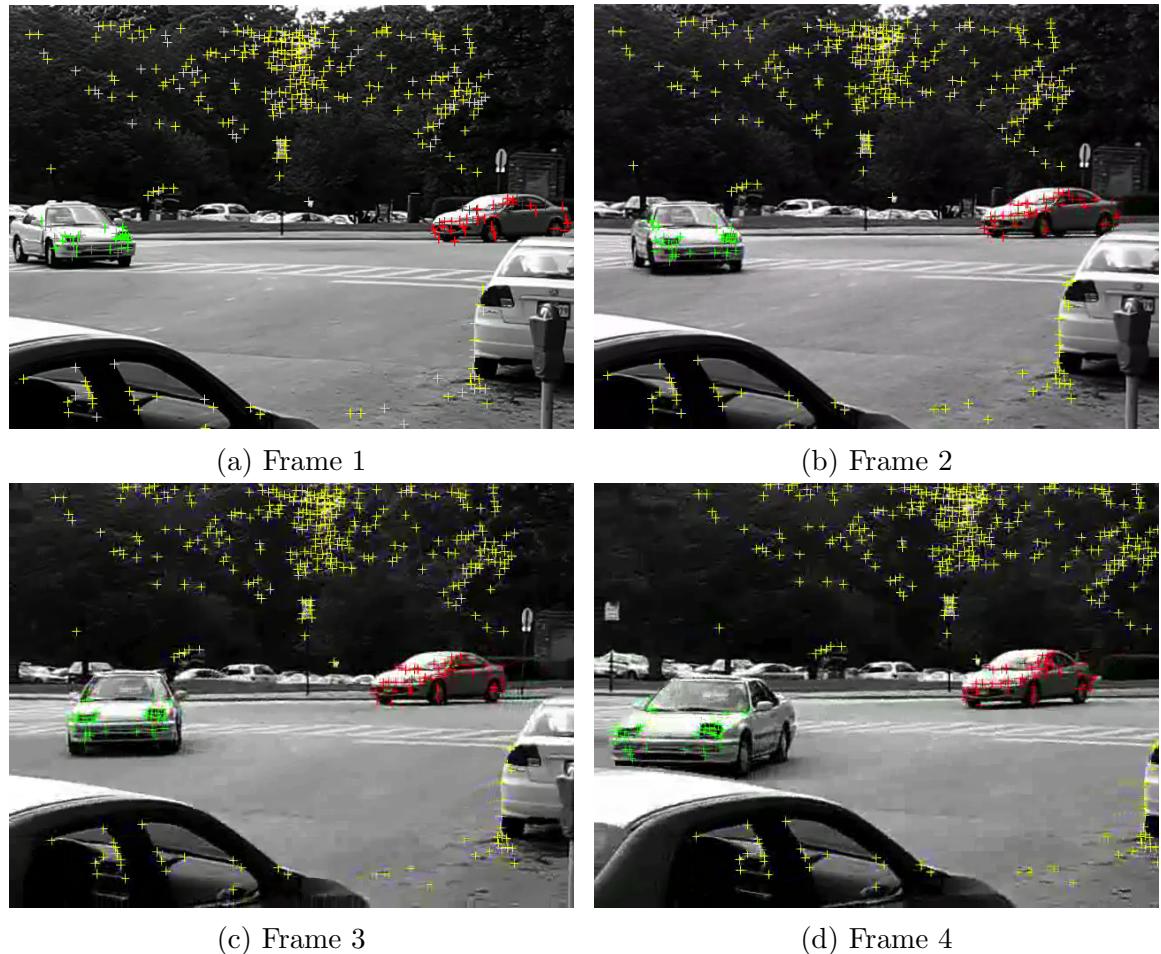


Figure 2.1: Point trajectories of two moving cars and a nonmoving background across 4 different frames. Each color corresponds to a different object (cluster).

point tracked across F frames so that $n = 2F$. More specifically,

$$\mathbf{X}^{(\ell)} = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1N_\ell} \\ y_{11} & y_{12} & \dots & y_{1N_\ell} \\ \vdots & \vdots & \ddots & \vdots \\ x_{F1} & x_{F2} & \dots & x_{FN_\ell} \\ y_{F1} & y_{F2} & \dots & y_{FN_\ell} \end{bmatrix}.$$

Consider a feature point $(X, Y, Z) \in \mathbb{R}^3$. Under the affine camera model this feature point is related to its projection on the image plane $(x, y) \in \mathbb{R}^2$ by

$$\begin{bmatrix} x \\ y \end{bmatrix} = \underbrace{\mathbf{K} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix}}_{\mathbf{A} \in \mathbb{R}^{2 \times 4}} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}.$$

Here, $\mathbf{K} \in \mathbb{R}^{2 \times 3}$ is the calibration matrix and (\mathbf{R}, \mathbf{t}) characterized the relative orientation of the image plane with respect to the world coordinates. As a result

$$\mathbf{X}^{(\ell)} = \underbrace{\begin{bmatrix} x_{11} & x_{12} & \dots & x_{1N_\ell} \\ y_{11} & y_{12} & \dots & y_{1N_\ell} \\ \vdots & \vdots & \ddots & \vdots \\ x_{F1} & x_{F2} & \dots & x_{FN_\ell} \\ y_{F1} & y_{F2} & \dots & y_{FN_\ell} \end{bmatrix}}_{\mathbf{A} \in \mathbb{R}^{2F \times 4}} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \\ \vdots \\ \mathbf{A}_F \end{bmatrix} \begin{bmatrix} X_1 & X_2 & \dots & X_{N_\ell} \\ Y_1 & Y_2 & \dots & Y_{N_\ell} \\ Z_1 & Z_2 & \dots & Z_{N_\ell} \\ 1 & 1 & \dots & 1 \end{bmatrix}.$$

This implies that $\text{rank}(\mathbf{X}^{(\ell)}) \leq 4$. Therefore, under the affine camera model the trajectories of feature points of a single rigid moving object will lie on a linear subspace (of dimension at most 4) of $\mathbb{R}^{n=2F}$. When there are multiple moving objects the set of all trajectories will lie in a union of linear subspaces of $\mathbb{R}^{n=2F}$. However, we do not know which trajectory belongs to which subspace. As a result motion segmentation of the trajectories is an instance of the subspace clustering problem.

2.2 Face clustering

It is known that different images of the face of a person shot from the same viewpoint but under varying illumination lie approximately on a low-dimensional subspace [131]. Thus, a set of images of different people under varying illumination lie close to a union of subspaces. The goal in face clustering is to cluster these images based on the identity of the people and is therefore an instance of subspace clustering.

2.3 Temporal segmentation of motion capture data

In this application we are given sensor measurements at multiple joints of the human body captured at different time instants. The goal is to segment the sensory data so that each cluster corresponds to the same activity. Here, each data point corresponds to a vector whose elements are the sensor measurements of different joints at a fixed time instant.

To demonstrate why temporal segmentation of motion capture data is an instance of the subspace clustering problem we use the Carnegie Mellon Motion Capture dataset (available at <http://mocap.cs.cmu.edu>), which contains 149 subjects performing several activities (data are provided in [253]). The motion capture system uses 42 markers per subject. We consider the data from subject 86 in the dataset, consisting of 15 different trials, where each trial comprises multiple activities. Figure 2.2 shows a few snapshots of each activity (walking, squatting, punching, standing, running, jumping, arms-up, and drinking) from trial 2. The right plot in Figure 2.2 shows the singular values of three of the activities in this trial. Notice that all the curves have a low-dimensional knee, showing that the data from each activity lie approximately in a low-dimensional subspace of the ambient space ($n = 42$ for all the motion capture data).

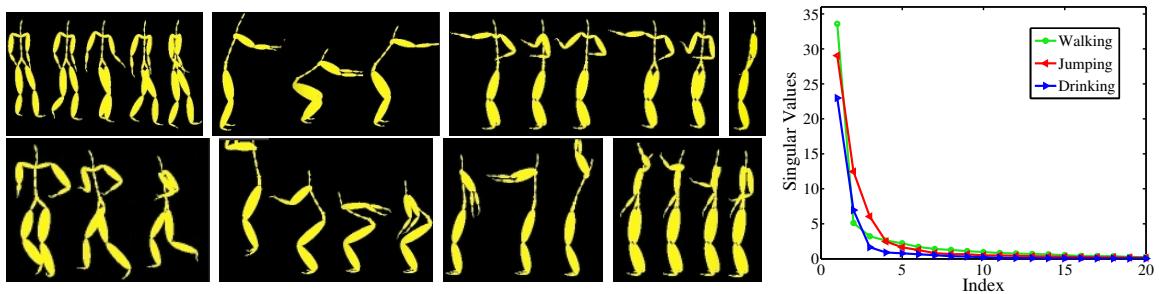


Figure 2.2: Left: eight activities performed by subject 86 in the CMU motion capture dataset: walking, squatting, punching, standing, running, jumping, arms-up, and drinking. Right: singular values of the data from three activities (walking, jumping, drinking) show that the data from each activity lie approximately in a low-dimensional subspace.

Chapter 3

Prior art in subspace clustering

In this chapter we review some existing approaches to subspace clustering. Following [233] we broadly classify existing subspace clustering techniques into four categories, namely, algebraic, iterative, statistical and spectral clustering based methods.

3.1 Generalized PCA (GPCA): an algebraic approach

As a representative of algebraic algorithms for subspace clustering, in this section we describe Generalized PCA (GPCA) [160, 234] which is an algebro-geometric subspace clustering technique. We shall mostly focus on the noise-free model for subspace clustering. However, throughout we shall explain briefly how the GPCA algorithm can be modified to tolerate moderate amounts of noise. Also, without loss of generality we consider the case where the data points are drawn from a union of linear subspaces, i.e. $\mu_\ell = \mathbf{0}$ for $\ell = 1, 2, \dots, L$. In the noise-free model, the same techniques can also be applied to handle affine subspaces by viewing an affine subspace of dimension d in \mathbb{R}^n as a linear subspace of dimension $d+1$ in \mathbb{R}^{n+1} . This is accomplished by lifting each data point $\mathbf{x} \in \mathbb{R}^n$ into its homogeneous coordinates $\begin{bmatrix} \mathbf{x}^T & 1 \end{bmatrix}^T \in \mathbb{R}^{n+1}$.

GPCA is an algebraic technique for subspace clustering. The key idea is to represent a union of L subspaces in \mathbb{R}^n with a set of homogeneous polynomials of degree

L in n variables and encode the subspaces bases as factors of these polynomials. It is possible to infer these polynomials given a sufficient number of sample points (in general position) from each subspace. In turn, a basis for the complement of each subspace can be deduced from the derivatives of these polynomials at a point in each of the subspaces. Finally, such points can be recursively selected via polynomial division. In the next sections we briefly explain these steps. To facilitate our exposition we will demonstrate each step on a simple example where we assume that we have data points drawn from two subspaces: a line $S_1 = \{\mathbf{x} \in \mathbb{R}^3 : x_1 = x_2 = 0\}$ and a plane $S_2 = \{\mathbf{x} \in \mathbb{R}^3 : x_3 = 0\}$.

3.1.1 Representing union of subspaces via homogeneous polynomials

Throughout we shall use

$$\mathbf{V}^{(\ell)} = [\mathbf{v}_{1\ell}, \mathbf{v}_{2\ell}, \dots, \mathbf{v}_{(n-d_\ell)\ell}] \in \mathbb{R}^{n \times (n-d_\ell)}$$

to denote a basis for the orthogonal complement of subspace S_ℓ (denoted by S_ℓ^\perp). Using these bases each subspace can be represented by a set of point obeying $n - d_\ell$ linear constraints, that is,

$$S_\ell = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{x}^T \mathbf{V}^{(\ell)} = \mathbf{0}\} = \{\mathbf{x} \in \mathbb{R}^n : \bigwedge_{k=1}^{n-d_\ell} (\mathbf{v}_{k\ell}^T \mathbf{x} = 0)\}.$$

We now demonstrate how the union of L subspaces can be associated with a set of polynomials of degree at most L . To this aim note that $\mathbf{x} \in S_1 \cup S_2 \cup \dots \cup S_L$ if and only if $\bigvee_{\ell=1}^L (\mathbf{x} \in S_\ell)$ holds. By De Morgan's law this is equivalent to

$$\bigvee_{\ell=1}^L (\mathbf{x} \in S_\ell) \Leftrightarrow \bigvee_{\ell=1}^L \bigwedge_{k=1}^{n-d_\ell} (\mathbf{v}_{k\ell}^T \mathbf{x} = 0) \Leftrightarrow \bigwedge_{\Pi} \bigvee_{\ell=1}^L (\mathbf{v}_{\ell\Pi(\ell)}^T \mathbf{x} = 0) \Leftrightarrow \bigwedge_{\Pi} \Leftrightarrow \bigwedge_{\Pi} (p_{\Pi}(\mathbf{x}) = 0), \quad (3.1.1)$$

where Π assigns one normal vector $\mathbf{v}_{\ell\Pi(\ell)}$ from each basis $\mathbf{V}^{(\ell)}$ and p_Π is a homogeneous polynomial of degree L in n variables.

The space of all homogenous polynomials of degree L in \mathbb{R}^n is a vector space of dimension $D_L(n) = \binom{L+n-1}{n-1}$. Polynomials in this vector space can be written as a linear combination of monomials $p(\mathbf{x}) = \mathbf{c}^T \boldsymbol{\nu}_L(\mathbf{x})$ where $\mathbf{c} \in \mathbb{R}^{D_L(n)}$ is the vector of coefficients and $\boldsymbol{\nu}_L : \mathbb{R}^n \rightarrow \mathbb{R}^{D_L(n)}$ is the Veronese map of degree L (all monomials placed in degree-lexicographic order). Thus a union of subspaces can be represented as the set of points satisfying a set of linear constraints of the form $\mathbf{c}^T \boldsymbol{\nu}_L(\mathbf{x}) = 0$.

On our running example we have

$$\begin{aligned} S_1 \cup S_2 &= \{\mathbf{x} : (x_1 = x_2 = 0) \vee (x_3 = 0)\} \\ &= \{\mathbf{x} : (x_1 x_3 = 0) \wedge (x_2 x_3 = 0)\}. \end{aligned}$$

Therefore, the two polynomials constraints representing our subspaces are

$$\begin{aligned} p_1(\mathbf{x}) &= x_1 x_3 = \mathbf{c}_1^T \boldsymbol{\nu}_L(\mathbf{x}) = 0 \\ p_2(\mathbf{x}) &= x_2 x_3 = \mathbf{c}_2^T \boldsymbol{\nu}_L(\mathbf{x}) = 0, \end{aligned}$$

where $\mathbf{c}_1 = [0 \ 0 \ 1 \ 0 \ 0 \ 0]^T$, $\mathbf{c}_2 = [0 \ 0 \ 0 \ 0 \ 1 \ 0]^T$, and $\boldsymbol{\nu}_L(\mathbf{x}) = [x_1^2 \ x_1 x_2 \ x_1 x_3 \ x_2^2 \ x_2 x_3 \ x_3^2]^T$.

3.1.2 Fitting polynomials to the data samples

In Section 3.1.1 we described how one can represent or encode union of subspaces via homogeneous polynomials. This representation will not be useful for subspace clustering unless we can infer these polynomials from sample data points. Let us assume the coefficients (in the space of polynomials) of $p_\Pi(\mathbf{x})$ in (3.1.1) is given by \mathbf{c}_Π , i.e. $p_\Pi(\mathbf{x}) = \mathbf{c}_\Pi^T \boldsymbol{\nu}_L(\mathbf{x})$. In mathematical terms, given sample data points $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$ from the union of subspaces the goal is to infer $\text{span}(\mathbf{c}_\Pi)$. We note

that for any $\mathbf{c} \in \text{span}(c_{\Pi})$ and its corresponding vanishing polynomial we have

$$p(\mathbf{x}_1) = p(\mathbf{x}_2) = \dots = p(\mathbf{x}_N) = 0 \quad \Rightarrow \quad \mathbf{c}^T [\boldsymbol{\nu}_L(\mathbf{x}_1) \ \ \boldsymbol{\nu}_L(\mathbf{x}_2) \ \ \dots \ \ \boldsymbol{\nu}_L(\mathbf{x}_N)] = \mathbf{0}^T.$$

Defining $\mathcal{V}_L(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n) = [\boldsymbol{\nu}_L(\mathbf{x}_1) \ \ \boldsymbol{\nu}_L(\mathbf{x}_2) \ \ \dots \ \ \boldsymbol{\nu}_L(\mathbf{x}_N)]$ the latter implies that

$$\text{span}(c_{\Pi}) \subset \mathcal{V}_L(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N).$$

Although we know that the coefficient vectors of vanishing polynomials lie in the left null space of $\mathcal{V}_L(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N)$, not every vector in the null space may correspond to a polynomial that vanishes on the subspaces. It turns out that as long as the subspaces obey some technical assumptions (which we shall explain in more detail in Section 3.1.6) and there are sufficiently many samples from each subspace in general position then we have

$$\text{span}(c_{\Pi}) = \mathcal{V}_L(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N). \quad (3.1.2)$$

Therefore, as soon as there are sufficient number of samples in general position from each subspace (3.1.2) allows us to find a basis $\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_m$ for $\text{span}(c_{\Pi})$ by the set of m left singular vectors of $\mathcal{V}_L(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N)$ associated with its m zero singular values. Therefore, we arrive at a basis of polynomials of degree L , $\{p_r\}_{r=1}^m$, that vanish on the L subspaces.

In the presence of moderate amount of noise we can still estimate the coefficient basis vectors $\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_m$ in a least-squares sense using singular vectors of $\mathcal{V}_L(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N)$ associated with its smallest singular values. However, the singular values are not exactly zero anymore so we need a method for estimating the parameter m . There are various heuristics in the model selection literature for this purpose; e.g. one popular heuristic is the eigen-gap heuristic where one tries to estimate m by seeing where the sharpest decrease is in the sorted singular values. Please see [132] for other approaches.

We shall now demonstrate this step on our running example. For this purpose let us assume that we have a nonzero point $\mathbf{w} \in S_1$ and three nonzero points $\mathbf{x}, \mathbf{y}, \mathbf{z} \in S_2$.

We have

$$\mathcal{V}_2(\mathbf{w}, \mathbf{x}, \mathbf{y}, \mathbf{z}) = \begin{bmatrix} 0 & x_1^2 & y_1^2 & z_1^2 \\ 0 & x_1x_2 & y_1y_2 & z_1z_2 \\ 0 & 0 & 0 & 0 \\ 0 & x_2^2 & y_2^2 & z_2^2 \\ 0 & 0 & 0 & 0 \\ w_3^2 & 0 & 0 & 0 \end{bmatrix}.$$

Assuming that no two of \mathbf{x} , \mathbf{y} and \mathbf{z} are collinear with each other the left null space of $\mathcal{V}_2(\mathbf{w}, \mathbf{x}, \mathbf{y}, \mathbf{z})$ has dimension 2 and is equal to $\text{span}(\mathbf{c}_1, \mathbf{c}_2)$ with $\mathbf{c}_1 = [0 \ 0 \ 1 \ 0 \ 0 \ 0]^T$ and $\mathbf{c}_2 = [0 \ 0 \ 0 \ 0 \ 1 \ 0]^T$. Therefore, given these sample points we conclude that the vanishing polynomials are

$$p_1(\mathbf{x}) = x_1x_3$$

$$p_2(\mathbf{x}) = x_2x_3.$$

3.1.3 Obtaining the subspace bases by differentiation

Now that we have explained how to estimate the vanishing polynomials from the sample data points we turn our attention to estimating the subspaces from these vanishing polynomials. We note that the zero set of each vanishing polynomial p_r is a surface in \mathbb{R}^n and therefore the gradient of p_r at a point $\mathbf{w}_\ell \in S_\ell$ ($\nabla p_r(\mathbf{w}_\ell)$), gives a vector normal to the surface. The surface in a neighborhood of $\mathbf{w}_\ell \in S_\ell$ is just the subspace S_ℓ so that $\nabla p_r(\mathbf{w}_\ell)$ gives a direction orthogonal to subspace S_ℓ . Now, by evaluating the derivative of the polynomials $\{p_r\}_{r=1}^m$ at the same point \mathbf{w}_ℓ we obtain a set of normal vectors that span the orthogonal complement of S_ℓ . More precisely, given a point $\mathbf{w}_\ell \in S_\ell$ we have

$$S_\ell^\perp = \text{span}(\nabla p_1(\mathbf{w}_\ell), \nabla p_2(\mathbf{w}_\ell), \dots, \nabla p_m(\mathbf{w}_\ell)). \quad (3.1.3)$$

Therefore, as soon as we know one point per subspace and the vanishing polynomials we can learn a basis for each of the subspaces. We can then proceed to label the rest of the data points by assigning them to their closest subspace (This is easy since we know a basis for each subspace).

On our running example assume $\mathbf{x} \in S_1$ and $\mathbf{y} \in S_2$. We have

$$\text{span}(\nabla p_1(\mathbf{x}), \nabla p_2(\mathbf{x})) = \text{span}\left(\begin{bmatrix} x_3 \\ 0 \\ x_1 \end{bmatrix}, \begin{bmatrix} 0 \\ x_3 \\ x_2 \end{bmatrix}\right) = \text{span}\left(\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}\right) = S_1^\perp.$$

We note that the penultimate equality follows from the fact that for $\mathbf{x} \in S_1$, $x_3 = 0$. Similarly,

$$\text{span}(\nabla p_1(\mathbf{y}), \nabla p_2(\mathbf{y})) = \text{span}\left(\begin{bmatrix} y_3 \\ 0 \\ y_1 \end{bmatrix}, \begin{bmatrix} 0 \\ y_3 \\ y_2 \end{bmatrix}\right) = \text{span}\left(\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}\right) = S_2^\perp.$$

3.1.4 Choosing one point per subspace by polynomial division

In the previous section we explained we can extract basis for each subspace provided we were given a point on each subspace as well as the vanishing polynomials. We now describe how to obtain a point for each subspace i.e. the points $\mathbf{w}_\ell \in S_\ell$ pertaining to (3.1.3). To this end note that for the first point we can choose just any random point as in the noiseless model it belongs to one of the subspaces. Using this point we can infer the corresponding subspace. However, in the presence of noise and outliers, a random selection may be far from the true subspaces. Ideally, we would like to choose a point which is closest to one of our subspaces. The lemma below provides us with such a measure.

Lemma 3.1.1 [234] Let $\tilde{\mathbf{x}}$ be the projection of $\mathbf{x} \in \mathbb{R}^n$ onto its closest subspace. The

Euclidean distance from \mathbf{x} to $\tilde{\mathbf{x}}$ is

$$\|\mathbf{x} - \tilde{\mathbf{x}}\|_{\ell_2} = L\sqrt{P(\mathbf{x})(DP(\mathbf{x})^T DP(\mathbf{x}))^\dagger P(\mathbf{x})^T} + \mathcal{O}(\|\mathbf{x} - \tilde{\mathbf{x}}\|_{\ell_2}^2),$$

where $P(\mathbf{x}) = [p_1(\mathbf{x}) \ p_2(\mathbf{x}) \ \dots p_m(\mathbf{x})] \in \mathbb{R}^{1 \times m}$, $DP(\mathbf{x}) = [\nabla p_1(\mathbf{x}) \ \nabla p_2(\mathbf{x}) \ \dots \nabla p_m(\mathbf{x})] \in \mathbb{R}^{n \times m}$, and \mathbf{A}^\dagger is the Moore-Penrose inverse of \mathbf{A} .

When dealing with noisy data Lemma 3.1.1 allows us to choose a point lying close to one of the subspaces and far from the other subspaces:

$$\mathbf{w}_L = \arg \min_{\mathbf{y} \in \mathcal{Y}} P(\mathbf{y})(DP(\mathbf{y})^T DP(\mathbf{y}))^\dagger P(\mathbf{y})^T.$$

Subsequently, we can find a basis for the subspace corresponding to \mathbf{w}_L (without loss of generality say S_L) as detailed in the previous sections.

Having picked one point, we would like to proceed by finding a point \mathbf{w}_{L-1} lying close to one of the remaining $L-1$ subspaces but far from S_L . We shall find a new set of polynomials that vanish on the set $S_1 \cup S_2 \cup \dots, S_{L-1}$ and then pick the next point using the same strategy employed for picking \mathbf{w}_L using the new set of polynomials. The next Lemma from [234] allows us to construct such polynomials via polynomial division. To state this lemma we need to first introduce some notation. Let $\tilde{\mathbf{c}}$ be the coefficients of a homogeneous polynomial of degree $L-1$ obtained by dividing $p(\mathbf{x}) = \mathbf{c}^T \boldsymbol{\nu}_L(\mathbf{x})$ by $\mathbf{b}^T \mathbf{x}$. It follows from standard calculus that

$$\tilde{\mathbf{c}}^T \mathbf{R}_L(\mathbf{b}) = \mathbf{c}^T,$$

where $\mathbf{R}_L(\mathbf{b}) \in \mathbb{R}^{D_{L-1}(n) \times D_L(n)}$ depends only on the vector \mathbf{b} .

Lemma 3.1.2 (Obtaining points by polynomial division) [234] *Let \mathcal{I}_L be (the space of coefficient vectors of) the set of polynomials vanishing on the L subspaces. If the noiseless data set \mathcal{X} is such that $\dim(\text{null}(\mathcal{V}_L(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N))) = \dim(\mathcal{I}_L)$, then the set of homogeneous polynomials of degree ($L-1$) that vanish on the algebraic set $\bigcup_{\ell=1}^{L-1} S_\ell$ is spanned by $\{\mathbf{c}_{L-1}^T \boldsymbol{\nu}_{L-1}(\mathbf{x})\}$, where the vectors of coefficients $\mathbf{c}_{L-1} \in \mathbb{R}^{D_{L-1}(n)}$*

must satisfy

$$\mathbf{c}_{L-1}^T R_L(\mathbf{b}) \mathcal{V}_L(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N) = \mathbf{0}^T \quad \text{for all } \mathbf{b} \in S_L^\perp.$$

Due to Lemma 3.1.2 above we can obtain a set of polynomials

$P_{L-1} = [p_{(L-1)1}(\mathbf{x}), p_{(L-1)2}(\mathbf{x}), \dots, p_{(L-1)m_{L-1}}(\mathbf{x})]$ which vanish on $\bigcup_{\ell=1}^{L-1} S_\ell$ from the intersection of the left null spaces of $R_L(\mathbf{b}) \mathcal{V}_L(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N)$ for all $\mathbf{b} \in S_L^\perp$. We can recursively repeat the same procedure to find a basis for the remaining subspaces.

We will now carry out these steps for our running example. For this purpose let us assume that we have a nonzero point $\mathbf{w} \in S_1$ and three nonzero points $\mathbf{x}, \mathbf{y}, \mathbf{z} \in S_2$. Say, we pick $\mathbf{w}_2 = \mathbf{z} \in S_2$ as our first point. We can now compute the normal basis to S_2 using $\text{DP}(\mathbf{w}_2)$ as $\mathbf{V}^{(2)} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$. We can pick a second point in S_1 which is not in S_2 by dividing the original polynomials by $(\mathbf{V}^{(2)})^T \mathbf{x}$ to obtain polynomials of degree 1:

$$p_{11}(\mathbf{x}) = \frac{p_1(\mathbf{x})}{(\mathbf{V}^{(2)})^T \mathbf{x}} = x_1 \quad \text{and} \quad p_{12} = \frac{p_2(\mathbf{x})}{(\mathbf{V}^{(2)})^T \mathbf{x}} = x_2.$$

Since these new polynomials vanish on S_1 and not S_2 the previous point selection procedure will now pick $\mathbf{w}_2 = \mathbf{w} \in S_1$, which in turn allows us to find the basis for S_1 .

3.1.5 The GPCA algorithm

In the previous sections we have discussed all the necessary pieces to introduce the GPCA algorithm. We have summarized all of these steps in Algorithm 1.

3.1.6 Some theory for GPCA

In the previous sections we outlined the GPCA algorithm. In this section we briefly review some theory for this algorithm. All the theory presented in this section is for the noiseless model and is mostly adapted from [160]. To the extent of our knowledge no analogous theory exists for any of the corruption models described in Section 1.1. We refer to [160, 197, 234] for some heuristic approaches for handling corruptions. We

Algorithm 1 GPCA: Generalized Principal Component Analysis [234]

Input: Data points $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$.

$$\text{Set } \mathcal{V}_L = [\boldsymbol{\nu}_L(\mathbf{x}_1) \ \boldsymbol{\nu}_L(\mathbf{x}_2) \ \dots \ \boldsymbol{\nu}_L(\mathbf{x}_N)] \in \mathbb{R}^{D_L(n) \times N}$$

for $\ell = L$ **to** 1 **do**

Solve $\mathbf{c}^T \mathcal{V}_\ell = \mathbf{0}$ to obtain a basis $\{\mathbf{c}_{r\ell}\}_{r=1}^{m_\ell}$ of $\text{null}(\mathcal{V}_\ell)$.

Set $P_\ell = [p_{\ell 1}(\mathbf{x}) \ p_{\ell 2}(\mathbf{x}) \ \dots \ p_{\ell m_\ell}(\mathbf{x})] \in \mathbb{R}^{1 \times m_\ell}$, where $p_{\ell r} = \mathbf{c}_{\ell r}^T \boldsymbol{\nu}_\ell(\mathbf{x})$ for $r = 1, 2, \dots, m_\ell$

Set

$$\mathbf{w}_\ell = \arg \min_{\mathbf{x} \in \mathcal{X}} P_\ell(\mathbf{x}) (DP_\ell(\mathbf{x})^T DP_\ell(\mathbf{x}))^\dagger P_\ell(\mathbf{x})^T,$$

$$\mathbf{U}^{(\ell)} = (\text{span}(DP_\ell(\mathbf{w}_\ell)))^\perp,$$

$$\mathcal{V}_{\ell-1} = [\mathbf{R}_\ell(\mathbf{U}_1^{(\ell)}) \mathcal{V}_\ell \ \mathbf{R}_\ell(\mathbf{U}_2^{(\ell)}) \mathcal{V}_\ell \ \dots \ \mathbf{R}_\ell(\mathbf{U}_{n-d_\ell}^{(\ell)}) \mathcal{V}_\ell], \mathbf{U}_r^{(\ell)} \text{ is the } r\text{th column of } \mathbf{U}^{(\ell)}.$$

end for

for $r = 1$ **to** N **do**

Assign point \mathbf{x}_r to subspace S_ℓ if $\ell = \arg \min_k \|(\mathbf{U}^{(k)})^T \mathbf{x}_r\|_{\ell_2}$.

end for

shall start with a few definitions.

For a nonempty subset \mathcal{L} of the index set $\{1, 2, \dots, L\}$, we define the intersection

$$S_{\mathcal{L}} = \bigcap_{\ell \in \mathcal{L}} S_\ell,$$

with dimension $d_{\mathcal{L}}$ and codimension $c_{\mathcal{L}} := n - d_{\mathcal{L}}$.

We now define the notion of transversal subspaces which is the property that subspaces are arranged in a manner that the dimensions of all intersections are as small as possible.

Definition 3.1.3 (transversal subspace arrangement) *A union of subspaces $S_1 \cup S_2 \cup \dots \cup S_L$ is called transversal if*

$$c_{\mathcal{L}} = \min \left(n, \sum_{\ell \in \mathcal{L}} c_\ell \right) \quad \text{for all nonempty } \mathcal{L} \subset \{1, 2, \dots, L\}.$$

We are now ready to present the main theoretical result for the GPCA algorithm based on [160].

Theorem 3.1.4 Assume the union of subspaces $S_1 \cup S_2 \cup \dots \cup S_L$ are transversal. Also, assume there are sufficiently many samples (in general position) from each subspace. Then there is a tractable method to compute $\{m_\ell\}_{\ell=1}^L$ in the GPCA algorithm from the data points. Furthermore, in the noiseless model the GPCA algorithm recovers the number of subspaces, their dimension and the basis for each subspace.

Remark 3.1.7 A precise lower bound on the number of samples required from each subspace is not understood. However, it is known that an overall number of $N > D_L(n) - 1$ is sufficient [234].¹

Remark 3.1.8 The algorithm that allows for choosing $\{m_\ell\}_{\ell=1}^L$ when there are sufficiently large number of samples is only tractable for small subspace dimensions. More precisely, the computational complexity is exponential in the subspace dimensions. Please see [85] for further details.

3.1.9 Pros and cons of GPCA

The main advantage of GPCA is that it handles the subspace clustering problem in a principled manner. Subspace clustering algorithms (similar to most clustering algorithms) are often based on nonconvex heuristics and a good understanding of these algorithms do not exist. The methodology developed in GPCA side steps these issues. The second advantage of GPCA is that it can handle intersection between subspaces automatically.

The main drawback of GPCA is that its complexity increases exponentially in terms of the parameters L and d_1, d_2, \dots, d_L (This is clear by noting that $D_L(n) \geq \left(\frac{L+n-1}{L}\right)^L$). Second, GPCA and its modifications are not robust to outliers and noise. Indeed, as GPCA is an algebraic scheme it is difficult to modify the algorithm to deal with noise and outliers in a principled manner. To give an example, the number and dimensions of the subspaces (which are need for GPCA to work properly) in the presence of noise are difficult to estimate. Third, while in principle GPCA can be

¹We should mention that even for this sufficient condition it is not clear how many points are required from each of the subspaces. That is how many of these points belong to S_1 , how many to S_2 , etc.

modified to deal with affine subspaces (as explained previously) this approach does not seem to work very well in the presence of noise. As a result of these issues, in practice the performance of GPCA degrades significantly as the number of subspaces increase. Finally, the theoretical understanding of GPCA even in the noiseless case is limited and restricted to transversal subspace arrangements. Furthermore, even the number of sample per subspace that is required for the algorithm to work (sampling complexity) is not well understood.

3.2 Iterative algorithms

A second class of algorithms is based on iterative refinements which are similar in spirit to the K-means algorithm. However, instead of using cluster centers to model the data as in K-means they use subspaces. Given an initial segmentation of the data which is typically chosen at random, these algorithms proceed by alternating between two steps: (1) find a fit to each group using classical PCA (2) assign each of the data points to the closest subspace. This is the basic idea behind the K-planes algorithm [47] which is designed for the union of multiple hyperplanes. The K-subspaces algorithm [6, 227] is a generalization of K-planes to affine subspaces. We will briefly describe the K-subspaces algorithm as a representative of iterative schemes.

3.2.1 The K-subspaces algorithm

K-subspaces assumes that the number of subspaces L and their dimensions d_1, d_2, \dots, d_L are known and aims to find the best union of affine subspaces that fit the data via the following nonconvex optimization problem

$$\begin{aligned} \min_{\boldsymbol{\mu}_\ell, \mathbf{U}^{(\ell)}, \boldsymbol{\beta}_{k\ell}, w_{k\ell}} \quad & \sum_{\ell=1}^L \sum_{k=1}^N w_{k\ell} \left\| \mathbf{y}_k - \boldsymbol{\mu}_\ell - \mathbf{U}^{(\ell)} \boldsymbol{\beta}_{k\ell} \right\|_{\ell_2}^2 \\ \text{subject to} \quad & w_{k\ell} \in \{0, 1\} \quad \text{and} \quad \sum_{\ell=1}^L w_{k\ell} = 1. \end{aligned} \tag{3.2.1}$$

Here, $\boldsymbol{\mu}_\ell \in \mathbb{R}^n$ and $\mathbf{U}^{(\ell)} \in \mathbb{R}^{n \times d_\ell}$ are optimization variables corresponding to the affine offset and basis for each of the unknown subspaces S_ℓ . Also, $\boldsymbol{\beta}_{k\ell} \in \mathbb{R}^{d_\ell}$ are optimization variables that correspond to the lower dimensional representation of point \mathbf{y}_k in the affine subspace S_ℓ . Furthermore, $w_{k\ell}$ are optimization variables used to characterize membership of the k th data point to subspace S_ℓ , i.e. $w_{k\ell} = 1$ if point \mathbf{y}_k originates from subspace S_ℓ and $w_{k\ell} = 0$ otherwise. Given subspace parameters $\boldsymbol{\mu}_\ell$ and $\mathbf{U}^{(\ell)}$ and the lower dimensional representations $\boldsymbol{\beta}_{k\ell}$, the optimal value in (3.2.1) is obtained by setting

$$w_{k\ell} = \begin{cases} 1 & \text{if } \ell = \arg \max_{r \in \{1, 2, \dots, L\}} \|\mathbf{y}_k - \boldsymbol{\mu}_r - \mathbf{U}^{(r)} \boldsymbol{\beta}_{k\ell}\|_{\ell_2} \\ 0 & \text{o.w.} \end{cases} \quad (3.2.2)$$

Given $w_{k\ell}$ the objective function in (3.2.1) separates into L independent optimization problems which can be easily solved by PCA. Thus, the K-subspace algorithm starts from a random initialization of the data points and alternates between the two steps above. As there are a finite number of ways in which points can be assigned to clusters, this algorithm is guaranteed to converge to a local minimum after a finite number of iterations.

3.2.2 Pros and cons of K-subspaces

The main advantage of K-subspaces is that it is very easy to implement as each iteration corresponds to essentially assigning points to different clusters and estimating the subspaces using PCA. Also, the formulation is very versatile and the objective function can be easily modified to handle changes in the model such as affine subspaces, outliers, etc. The main drawback is that the objective function is nonconvex and depending on the initialization scheme the iterative updates may converge to a local minimum rather than the global optimum. Furthermore, even though there is some literature on possible initialization schemes for this algorithm (e.g. [10, 252]), in general it is not clear how K-subspaces should be initialized. Indeed, in practice many random initializations are required for the algorithm to reach a suitable estimate. Another major drawback is that this algorithm requires knowledge of the

number of subspaces and their dimensions. Finally, there is no theoretical analysis of the sampling/computational complexity of this algorithm.

3.3 Statistical algorithms

In this section we shall briefly review three statistical approaches to subspace clustering. The basic idea behind these approaches is to assume prior probabilistic models for the distribution of the data and then try to provide estimates for the subspaces based on a Maximum Likelihood (ML) estimation of the unknown parameters of the corresponding density function.

3.3.1 Mixture of probabilistic PCA

The Mixture of Probabilistic PCA (MPPCA) approach proposes a statistical mixture model to represent data lying near a union of subspaces. More specifically, the probabilistic model has the following density function

$$f(\mathbf{x}; \{\pi_\ell, \boldsymbol{\mu}_\ell, \boldsymbol{\Sigma}_\ell\}_{\ell=1}^L) = \sum_{\ell=1}^L \pi_\ell G\left(\mathbf{x}; \boldsymbol{\mu}_\ell, \mathbf{U}^{(\ell)} (\mathbf{U}^{(\ell)})^T + \sigma_\ell^2 \mathbf{I}\right), \quad \sum_{\ell=1}^L \pi_\ell = 1.$$

Here, $G(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the probability density function of an n -dimensional Gaussian random variable with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ and π_ℓ represents the probability that a point is drawn from the subspace S_ℓ . Also, we have used the short-hand $\boldsymbol{\Sigma}_\ell = \mathbf{U}^{(\ell)} (\mathbf{U}^{(\ell)})^T + \sigma_\ell^2 \mathbf{I}$. We note that each component in the mixture represents points distributed uniformly on the subspace but corrupted with Gaussian noise.

Given the data points one can try to maximize the likelihood of this distribution to get an estimate of the parameters of this model, i.e. $\pi_\ell, \sigma_\ell, \boldsymbol{\mu}_\ell$, and $\mathbf{U}^{(\ell)}$. That is, try to maximize

$$\mathcal{L}\left(\{\pi_\ell, \boldsymbol{\mu}_\ell, \boldsymbol{\Sigma}_\ell\}_{\ell=1}^L; \mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N\right) = \sum_{k=1}^N \log\left(f(\mathbf{y}_k; \{\pi_\ell, \sigma_\ell, \boldsymbol{\mu}_\ell, \boldsymbol{\Sigma}_\ell\}_{\ell=1}^L)\right).$$

This objective is nonconvex. A popular heuristic is to use the Expected Maximization

(EM) algorithm [84]. Given estimates $\{\bar{\pi}_\ell, \bar{\sigma}_\ell, \bar{\mu}_\ell, \bar{\Sigma}_\ell\}_{\ell=1}^L$ for the model parameters the E-step of the EM algorithm estimates the probability that points \mathbf{y}_k originates from subspace S_ℓ via

$$\tilde{p}_{k\ell} = \frac{\bar{\pi}_\ell G(\mathbf{y}_k; \bar{\mu}_\ell, \bar{\Sigma}_\ell)}{f(\mathbf{y}_k; \{\bar{\pi}_\ell, \bar{\mu}_\ell, \bar{\Sigma}_\ell\}_{\ell=1}^L)}.$$

The M-step uses these probability estimates to update subspace parameters π_ℓ , μ_ℓ and Σ_ℓ via

$$\tilde{\pi}_\ell = \frac{1}{N} \sum_{k=1}^N \tilde{p}_{k\ell}, \quad \tilde{\mu}_\ell = \frac{1}{N\tilde{\pi}_\ell} \sum_{k=1}^N \tilde{p}_{k\ell} \mathbf{y}_k, \quad \text{and} \quad \tilde{\Sigma}_\ell = \frac{1}{N\tilde{\pi}_\ell} \sum_{k=1}^N \tilde{p}_{k\ell} (\mathbf{y}_k - \tilde{\mu}_\ell)(\mathbf{y}_k - \tilde{\mu}_\ell)^T.$$

The E and M steps are iterated until the algorithm converges to a local maximum of the likelihood. The subspace parameters σ_ℓ and $\mathbf{U}^{(\ell)}$ are calculated from the final estimate for Σ_ℓ by Singular Value Decomposition (SVD).

We note that MMPCA can be viewed as a probabilistic version of K-subspaces that uses soft assignments $p_{k\ell} \in [0, 1]$ in lieu of hard assignments $w_{k\ell} \in \{0, 1\}$. Thus, MMPCA and K-subspaces are of a similar nature and the benefits and drawbacks of both algorithms are similar. Therefore, we refrain from mentioning them here again and refer the reader to Section 3.2.2.

3.3.2 Agglomerative lossy compression

Agglomerative Lossy Compression (ALC) [158] uses ideas from coding and lossy compression to segment multivariate mixed data that are distributed around a union of subspaces. Similar to other statistical approaches for subspace clustering ALC is also based on a generative model for the data. In particular, it is assumed that the data is drawn from a mixture of multiple Gaussian-like groups which may have significantly different, anisotropic, and even near degenerate covariances. ALC aims to find the optimal segmentation of the data, which results in the shortest coding length needed to fit the points with a mixture of degenerate Gaussians up to a given distortion.

A lossy coding scheme maps a set of vectors $\mathcal{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N \in \mathbb{R}^n\}$ to a binary

sequence such that the original vectors can be recovered up to a certain distortion level $\mathbb{E}[\|\mathbf{y} - \hat{\mathbf{y}}\|_{\ell_2}^2] \leq \epsilon^2$. We use $CL(\mathcal{Y})$ to denote the length of the smallest such binary sequence. For a given family of distributions, tools from information theory allow us to characterize the CL function and the corresponding coding scheme. In particular, when the data are i.i.d. samples from a zero-mean multivariate Gaussian distribution $\mathcal{N}(\mathbf{0}, \Sigma)$, using information theory we can approximate $CL(\mathcal{Y})$ by

$$CL(\mathcal{Y}) \approx \frac{n + |\mathcal{Y}|}{2} \log_2 \det \left(\mathbf{I} + \frac{n}{\epsilon^2} \Sigma \right).$$

Here, $|\mathcal{Y}|$ denotes the cardinality of the set \mathcal{Y} . The covariance matrix Σ can be estimated by the empirical covariance matrix $\hat{\Sigma} = \frac{1}{|\mathcal{Y}|} \sum_{\mathbf{y} \in \mathcal{Y}} \mathbf{y} \mathbf{y}^T$. Thus, we have

$$CL(\mathcal{Y}) \approx \frac{n + |\mathcal{Y}|}{2} \log_2 \det \left(\mathbf{I} + \frac{n}{\epsilon^2 |\mathcal{Y}|} \sum_{\mathbf{y} \in \mathcal{Y}} \mathbf{y} \mathbf{y}^T \right).$$

Given a set of points \mathcal{Y} there are multiple ways to code them. One alternative is to view all of the points as originating from a single Gaussian source and code \mathcal{Y} with distortion ϵ^2 via $CL(\mathcal{Y})$ bits. Another alternative is to code \mathcal{Y} as the union of multiple (disjoint) groups $\mathcal{Y} = \mathcal{Y}_1 \cup \mathcal{Y}_2 \cup \dots \cup \mathcal{Y}_s$. For example, the second alternative maybe more efficient if the samples are actually drawn from a mixture of Gaussian distributions or subspaces. If each group is coded separately, then the total number of bits needed is

$$CL(\mathcal{Y}_1 \cup \mathcal{Y}_2 \cup \dots \cup \mathcal{Y}_s) = \sum_{r=1}^s CL(\mathcal{Y}_r) - |\mathcal{Y}_r| \log_2 \left(\frac{|\mathcal{Y}_r|}{|\mathcal{Y}|} \right).$$

The extra term $- \sum_{r=1}^s |\mathcal{Y}_r| \log_2 \left(\frac{|\mathcal{Y}_r|}{|\mathcal{Y}|} \right)$ is the number of bits needed to losslessly code the membership of the points into their respective groups (Please see [80] for further details). To code the data points as succinctly as possible we wish to find a partition that minimizes the overall code length:

$$\underset{s, \mathcal{Y}_1, \mathcal{Y}_2, \dots, \mathcal{Y}_s}{\text{minimize}} \quad \sum_{r=1}^s \frac{n + |\mathcal{Y}_r|}{2} \log_2 \det \left(\mathbf{I} + \frac{n}{\epsilon^2 |\mathcal{Y}_r|} \sum_{\mathbf{y} \in \mathcal{Y}_r} \mathbf{y} \mathbf{y}^T \right) - |\mathcal{Y}_r| \log_2 \left(\frac{|\mathcal{Y}_r|}{|\mathcal{Y}|} \right). \quad (3.3.1)$$

The optimization in (3.3.1) is intractable as it involves optimization over all possible partitioning of the data. Nevertheless, ALC provides a greedy heuristic for this optimization problem by using an agglomerative clustering approach in a bottom-up fashion. At the beginning every sample is treated as a cluster. In subsequent iterations two clusters \mathcal{Y}_1 and \mathcal{Y}_2 are chosen so that merging them results in the greatest decrease in the coding length. The algorithm terminates when merging any pair of groups does not decrease the coding length. This merging technique is detailed in Algorithm 2.

Algorithm 2 Agglomerative merging technique in ALC

Input: Data points $\mathcal{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N\}$ and a distortion ϵ^2 .

Initialize $\mathcal{W} := \{\mathcal{W} = \{\mathbf{y}\} | \mathbf{y} \in \mathcal{Y}\}$.

while $|\mathcal{W}| > 1$ **do**

choose distinct sets $\mathcal{W}_1, \mathcal{W}_2 \in \mathcal{W}$ such that

$$CL(\mathcal{W}_1 \cup \mathcal{W}_2) - CL(\mathcal{W}_1, \mathcal{W}_2)$$

is minimal.

if $CL(\mathcal{W}_1 \cup \mathcal{W}_2) - CL(\mathcal{W}_1, \mathcal{W}_2) \geq 0$ **then**

break;

else

$\mathcal{W} := (\mathcal{W} \setminus \{\mathcal{W}_1, \mathcal{W}_2\}) \cup \{\mathcal{W}_1 \cup \mathcal{W}_2\}$.

end if

end while

The major drawback of ALC is that it is a greedy approach for a nonconvex optimization scheme and there is no guarantee that this algorithm can reach the global minimum and does not get trapped in a local minima. The main advantage of ALC is that it can handle outliers rather well empirically.

3.3.3 Random sampling consensus

Random Sampling Consensus (RANSAC) [108] is a paradigm, for fitting a model to a cloud of points in a manner which is robust to outliers. Assume we are trying to fit a model containing d parameters. RANSAC operates by randomly sampling d points from the data, fits the model using these points and then computes the residual of

each of the data points with respect to the ideal fit. Using the size of these residuals RANSAC rejects some data points as outliers and deems the rest to be inliers. This procedure is repeated on another randomly d data points until there are no further data points.

RANSAC can be applied to subspace clustering when all subspaces have equal dimensions d . The model to be fitted in this case is one of the subspaces and all points from the other subspaces are the outliers. RANSAC proceeds in a greedy fashion by fitting one subspace at a time. First a set of points is chosen and then a subspace is fitted to the inlier set. Then the inlier points are removed and the process is repeated, each time fitting a new subspace. After all the subspaces have been found, then RANSAC assigns each of the remaining data points to its closest subspace or identifies them as outliers.

The main advantage of RANSAC is that it is inherently robust to outliers. The main drawback of RANSAC is that its performance decreases significantly as the number of subspaces increases. The reason is that the probability that most random points that are selected in the first batch will belong to the same subspace decreases rapidly as the number of subspaces increases. Another drawback is that it requires all of the subspaces to have the same dimension.

3.4 Spectral clustering-based algorithms

Spectral clustering algorithms are among the most widely used clustering algorithms in modern data analysis. In this section we shall briefly explain how spectral clustering operates and then explain different algorithms that use this approach for the purpose of subspace clustering.

Given a set of data points $\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N$ and some measure of similarity $w_{ij} \geq 0$ between each pair of data points \mathbf{y}_i and \mathbf{y}_j , the intuitive goal of spectral clustering is to divide the data points into several groups such that points in the same group are similar and points in different groups are dissimilar to each other. This similarity often comes in the form of a weighted similarity graph $\mathcal{G}(\mathbf{W})$, where each vertex represents a data point and each edge the similarity between two data points; the

higher the edge weights w_{ij} (ij th entry of the matrix \mathbf{W}), the more similar the nodes i and j , and the larger the value w_{ij} . We will also define the degree of the i th node of the similarity graph by $d_i = \sum_{j=1}^N w_{ij}$. For the purposes of clustering we would like our affinity matrix to be a permutation of a block diagonal matrix so that nodes that belong to the same cluster have some similarity but node belonging to two different clusters are not similar at all.

The main tool for spectral clustering is the graph Laplacian matrix. In this thesis we shall work with the normalized graph Laplacian defined by

$$\mathcal{L} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}} \mathbf{W} \mathbf{D}^{-\frac{1}{2}}.$$

Here \mathbf{D} is a diagonal matrix equal to $\text{diag}(d_1, d_2, \dots, d_N)$ with obvious notation. The reason the graph Laplacian is useful for clustering is due to the following theorem which shows that when the graph has multiple connected components the eigenvalues of the Laplacian matrix acts as an indicator for each of the connected components.

Theorem 3.4.1 (properties of the Laplacian, [236]) *The normalized Laplacian satisfies the following properties:*

1. for every $\mathbf{v} \in \mathbb{R}^n$ we have

$$\mathbf{v}^* \mathcal{L} \mathbf{v} = \frac{1}{2} \sum_{i,j=1}^n w_{ij} \left(\frac{v_i}{\sqrt{d_i}} - \frac{v_j}{\sqrt{d_j}} \right)^2.$$

2. 0 is an eigenvalue of \mathcal{L} with eigenvector $\mathbf{D}^{\frac{1}{2}} \mathbf{1}$ where $\mathbf{1}$ is the all one vector.
3. \mathcal{L} is positive semi-definite and has N non-negative real-valued eigenvalues $0 = \lambda_1 \leq \dots \leq \lambda_N$.
4. The multiplicity L of the eigenvalue 0 of \mathcal{L} equals the number of connected components $\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_L$ in the graph. The eigenspace of 0 is spanned by the vectors $\mathbf{D}^{\frac{1}{2}} \mathbf{1}_{\mathcal{G}_i}$

We now have all the elements to state the most common spectral clustering algorithm detailed in Algorithm 3.

Algorithm 3 Spectral Clustering (SpecClust(\mathbf{W})) [189]

Input: Similarity matrix $\mathbf{W} \in \mathbb{R}^{N \times N}$ and number L of clusters to construct.

1. Compute the normalized Laplacian \mathcal{L} .
2. Compute the first L eigenvectors $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_L$ of \mathcal{L} .
3. Let $\mathbf{U} \in \mathbb{R}^{N \times L}$ be the matrix containing the vectors $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_L$ as columns.
4. Form the matrix $\tilde{\mathbf{U}} \in \mathbb{R}^{N \times L}$ from \mathbf{U} by normalizing the rows to norm 1, that is set $\tilde{u}_{ij} = u_{ij} / (\sum_k u_{ik}^2)^{1/2}$.
5. For $i = 1, 2, \dots, N$, let $\tilde{\mathbf{u}}_i \in \mathbb{R}^L$ be the vector corresponding to the i -th row of $\tilde{\mathbf{U}}$.
6. Cluster the points $\{\tilde{\mathbf{u}}_1, \tilde{\mathbf{u}}_2, \dots, \tilde{\mathbf{u}}_N\}$ with the K-means algorithm into clusters $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_L$.

Output: Clusters $\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_L$ with $\mathcal{G}_i = \{j | \tilde{\mathbf{u}}_j \in \mathcal{C}_i\}$.

The main challenge in applying spectral clustering techniques to subspace clustering is to define a good affinity matrix. The main reason is that subspace clustering is a nonstandard clustering problem in which neighbors are not close according to a predefined notion of metric but rather belong to the same lower dimensional structure (two points could be very close to each other but lie in different subspaces). In what follows we review some of the methods for building a similarity graph for points on a union of subspaces. Before we start explaining these algorithms we will begin with three definitions which are going to be useful throughout this discussion.

Definition 3.4.2 (independent subspaces) *The subspaces S_1, S_2, \dots, S_L are said to be independent if and only if $\sum_{\ell=1}^L \dim(S_\ell) = \dim(\oplus_\ell S_\ell)$ where \oplus is the direct sum.*

Definition 3.4.3 (block diagonal affinity matrix) *We will say the affinity (similarity) matrix \mathbf{W} is block diagonal if it is such that*

$$\mathbf{W}_{ij} = 0 \quad \text{if points } i \text{ and } j \text{ are in different subspaces.}$$

The following definitions capture notions of similarity/affinity between two subspaces.

Definition 3.4.4 *The principal angles $\theta_{k,\ell}^{(1)}, \dots, \theta_{k,\ell}^{(d_k \vee d_\ell)}$ between two subspaces S_k*

and S_ℓ of dimensions d_k and d_ℓ , are recursively defined by

$$\cos(\theta_{k\ell}^{(i)}) = \max_{\mathbf{y} \in S_k} \max_{\mathbf{z} \in S_\ell} \frac{\mathbf{y}^T \mathbf{z}}{\|\mathbf{y}\|_{\ell_2} \|\mathbf{z}\|_{\ell_2}} = \frac{\mathbf{y}_i^T \mathbf{z}_i}{\|\mathbf{y}_i\|_{\ell_2} \|\mathbf{z}_i\|_{\ell_2}}.$$

with the orthogonality constraints $\mathbf{y}^T \mathbf{y}_j = 0$, $\mathbf{z}^T \mathbf{z}_j = 0$, $j = 1, \dots, i-1$.

Alternatively, if the columns of $\mathbf{U}^{(k)}$ and $\mathbf{U}^{(\ell)}$ are orthobases, then the cosine of the principal angles are the singular values of $\mathbf{U}^{(k)T} \mathbf{U}^{(\ell)}$. We write the smallest principal angle as $\theta_{k\ell} = \theta_{k\ell}^{(1)}$ so that $\cos(\theta_{k\ell})$ is the largest singular value of $\mathbf{U}^{(k)T} \mathbf{U}^{(\ell)}$.

3.4.1 Cosine-based affinity

A natural way to measure similarity of two points is via collinearity. This leads to the following measure of similarity (first used in [145]) which test how close the data points are to each other in an angular sense

$$w_{ij} = \left(\frac{|\mathbf{y}_i^* \mathbf{y}_j|}{\|\mathbf{y}_i\|_{\ell_2} \|\mathbf{y}_j\|_{\ell_2}} \right)^\alpha.$$

Here, $\alpha \geq 1$ is the parameter that tunes the sharpness of the affinity between two points (another way to tune this sharpness is to use $\alpha = 1$ but set the similarities below a certain thresholds to zero). This notion of similarity would be ideal for clustering points along a union of lines. However, it is still an intuitive measure to use for linear subspaces as well. It is easy to show that when the subspaces are orthogonal to each other the above affinity matrix will be block diagonal. Thus, in the absence of noise, the above affinity can be used to obtain perfect segmentation of the data by spectral clustering.

While Cosine-based affinity is a natural, simple and fast method for subspace clustering it has some major drawbacks. First, it only applies to linear subspaces. Second, even when the data points contain no form of corruption it is only guaranteed to work for orthogonal subspaces.² Third, these algorithms tend to be extremely

²We note that thresholded variations of these algorithms have been recently analyzed and shown to work in less restrictive regimes [127, 128]. However, the choice of the threshold depends on

sensitive to the choice of the parameters (α or the choice of the threshold). Finally, there is no natural way to modify these algorithm in the presence of huge corruptions such as sparse outliers. As a result these class of algorithms do not usually work very well in practice. Indeed as we demonstrate in Sections 6.4 and 6.5 their performance is inferior to most commonly used subspace clustering techniques.

3.4.2 Factorization-based affinity

Factorization-based subspace clustering [45, 79, 113] are among the first subspace clustering algorithms. The intuition behind these class of affinity matrices is similar to the cosine-based affinity. However, instead of using the dot product between the data points the dot product is between some features of the data points. In particular, the features used are based on the SVD of the data points. Let $Y = U\Sigma V^T$ be the SVD of the data points \mathbf{Y} . Then the affinity is defined as

$$\mathbf{W} = |\mathbf{V}\mathbf{V}^T|,$$

where the absolute value is applied entry by entry. The rows of \mathbf{V} are the feature vectors used in lieu of the actual data points.

Factorization-based affinity suffers from the same drawbacks as cosine-based affinity. However, in comparison this form of affinity is much more stable and performs much better on real data. Also, there are many variations of these type of algorithms that further boost their performance on real data [45, 79, 133, 137–139, 245]. However, these modifications often come at the cost of additional tuning parameters.

3.4.3 Local subspace affinity and spectral local best-fit flats

Local Subspace Affinity (LSA) [247] and Spectral Local Best-fit Flats (SLBF) [252] are two algorithms that use the observation that a point and its nearest neighbors (NNs) often originate from the same subspace. Thus, corresponding to each data

problem parameters such as the dimension of the subspaces, etc. Furthermore, the required number of samples required per subspace is exponential in terms of the dimension of the subspace.

point \mathbf{y}_i one can fit a local subspace \hat{S}_i via PCA using that point and its K nearest neighbors (In practice, if \mathbf{y}_i originates from subspace S_ℓ with dimension d_ℓ then we choose $K \geq d_\ell$). One expects the local subspaces of two points to be similar when they originate from the same subspace and rather different when they originate from two different subspaces. Thus one can use some measure of distance between the local subspaces of two points as the affinity between these two points. Both LSA and SLBF follow this strategy. The difference between these three algorithms is threefold: (1) how the number of K NNs and dimensions of the local subspaces is determined, (2) how the subspaces are estimated and (3) how the affinity measure between the subspaces are calculated.

LSA assumes that K is specified by the user and finds the K NNs by using the angles between data points. The subspace dimension is determined using model selection techniques and the subspaces is fitted by PCA. We note that LSA can only handle linear subspaces. SLBF determines both the number of K_i NNs and the estimated subspace \hat{S}_i for each data point \mathbf{y}_i automatically by search for the smallest K_i that minimizes a certain fitting error.

LSA computes the affinity matrix via

$$\mathbf{W}_{ij} = \exp\left(-\sum_{r=1}^{\min(d_i, d_j)} \sin^2(\theta_{ij}^{(r)})\right),$$

where $\theta_{ij}^{(r)}$ is the r th principal angle between the estimated subspaces \hat{S}_i and \hat{S}_j . As mentioned earlier, this only applies to linear subspaces. SLBF uses the affinity

$$\mathbf{W}_{ij} = \exp\left(-\frac{\sqrt{\text{dist}(\mathbf{y}_i, \hat{S}_j)\text{dist}(\mathbf{y}_j, \hat{S}_i)}}{2\sigma_i^2}\right) + \exp\left(-\frac{\sqrt{\text{dist}(\mathbf{y}_i, \hat{S}_j)\text{dist}(\mathbf{y}_j, \hat{S}_i)}}{2\sigma_j^2}\right),$$

where σ_i is an estimate of how well the subspace \hat{S}_i fits point \mathbf{y}_i and its K_i -NNs and $\text{dist}(\mathbf{y}, S)$ is the Euclidean distance of point \mathbf{y} from subspace S .

The main advantage of both these algorithms is that they are relatively simple and fast to implement. Also a main advantage of SLBF is that it can automatically handle affine subspaces in a natural way. The main issue with both of these algorithms is

that there is not much theory explaining why these algorithms work. Also, similar to cosine-based affinity the parameters involved in LSA and SLBF make the algorithm sensitive to choice of parameters d_i and K_i . However, these algorithms are much less sensitive to these parameters in comparison to cosine-based affinities because of the use of the local subspaces. We also note that this problem is less prominent in SLBF due to the automatic choice of these quantities.

3.4.4 Spectral curvature clustering

The idea behind multiway clustering techniques such as [7, 73, 118] is to consider all $d+2$ combinations of the data points and try to somehow measure how likely is it that these points originate from the same subspace, and then use this measure to define an affinity between two points. We shall discuss Spectral Curvature Clustering (SCC) [73] as a representative of multiway clustering techniques.

Let $\mathbf{Y}_{d+2} \in \mathbb{R}^{n \times (d+2)}$ be a randomly chosen submatrix of size $n \times (d+2)$ from the data matrix \mathbf{Y} . The affinity used by SCC is based on the concept of polar curvature of these $d+2$ points. This value is zero when all the points reside exactly on the same subspace. While other functions of these $(d+2)$ may also have this property (e.g. volume of the $d+2$ points denoted by $\text{vol}(\mathbf{Y}_{d+2})$), the reason this particular function is useful is that it is invariant to data transformations such as scaling of the $d+2$ points. Let $(i_1, i_2, \dots, i_{d+2}) \in \{1, 2, \dots, N\}^{(d+2)}$, the multiway affinity tensor $\mathcal{W} \in \mathbb{R}^{n^{(d+2)}}$ is defined via its elements as follows

$$\mathcal{W}_{i_1, i_2, \dots, i_{d+2}} = \exp \left(-\frac{1}{2\sigma^2} \text{diam}^2(\mathbf{Y}_{d+2}) \sum_{r=1}^{d+2} \frac{((d+1)!)^2 \text{vol}^2(\mathbf{Y}_{d+2})}{\prod_{s=r}^{d+2} \|\mathbf{y}_{i_r} - \mathbf{y}_{i_s}\|_{\ell_2}^2} \right),$$

where $\text{diam}(\mathbf{Y}_{d+2})$ is the diameter of \mathbf{Y}_{d+2} . The pairwise affinity matrix is defined as

$$\mathbf{W}_{ij} = \sum_{i_2, i_3, \dots, i_{d+1} \in \{1, 2, \dots, N\}} \mathcal{W}_{i, i_2, \dots, i_{d+2}} \mathcal{W}_{i_1, i_2, \dots, j}.$$

Compared to the affinities we have discussed so far which are in some sense local affinity measures (the affinity is defined based on points close by or a pair of points)

multiway affinities are more global. Therefore, one expects them to yield more accurate clustering results. Indeed, as we will demonstrate in Section 6.4 for some applications this is certainly the case. Another advantage is that it can easily handle both affine and linear subspaces. The main drawback of SCC is its computational complexity as it requires computing and summing over $\mathcal{O}(N^{(d+2)})$ of \mathcal{W} which is exponential in the dimension of the subspaces so that the algorithm is intractable for large subspace dimensions. Another drawback is that it requires all the subspaces to be of equal dimension d . While there is some theory explaining why SCC succeeds for subspace clustering [72] this theory has its limitations. For example, the sampling complexity (or the number of points required per subspace as a function of subspace dimension) seems far from optimal.

Chapter 4

Robust subspace clustering

In this chapter we introduce our methodology through heuristic arguments confirmed by numerical experiments. In the next chapter (chapter 5) we provide theoretical guarantees about the algorithms discussed here. From now on, we arrange the N observed data points as columns of a matrix $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_N] \in \mathbb{R}^{n \times N}$. With obvious notation, $\mathbf{Y} = \mathbf{X} + \mathbf{Z}$.

4.1 The SSC scheme

We describe the approach in [96], which follows a three-step procedure:

- I Compute a similarity¹ matrix \mathbf{W} encoding similarities between sample pairs as to construct a weighted graph \mathcal{G} .
- II Construct clusters by applying spectral clustering techniques (e.g. [189]) to \mathcal{G} .
- III Apply PCA to each of the clusters.

The novelty in [96] concerns step I, the construction of the affinity matrix. Interestingly, similar ideas were introduced earlier in the statistics literature for the purpose of graphical model selection [167]. Now the work [96] of interest here is

¹We use the terminology similarity graph or matrix instead of affinity matrix as not to overload the word ‘affinity’.

mainly concerned with the noiseless situation in which $\mathbf{Y} = \mathbf{X}$ and the idea is then to express each column \mathbf{x}_i of \mathbf{X} as a sparse linear combination of all the other columns. The reason is that under any reasonable condition, one expects that the *sparsest* representation of \mathbf{x}_i would only select vectors from the subspace in which \mathbf{x}_i happens to lie in. Applying the ℓ_1 norm as the convex surrogate of sparsity leads to the following sequence of optimization problems

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^N} \|\boldsymbol{\beta}\|_{\ell_1} \quad \text{subject to} \quad \mathbf{x}_i = \mathbf{X}\boldsymbol{\beta} \text{ and } \beta_i = 0. \quad (4.1.1)$$

Here, β_i denotes the i th element of $\boldsymbol{\beta}$ and the constraint $\beta_i = 0$ removes the trivial solution that decomposes a point as a linear combination of itself. Collecting the outcome of these N optimization problems as columns of a matrix \mathbf{B} , [96] sets the $N \times N$ similarity matrix \mathbf{W} to be $W_{ij} = |B_{ij}| + |B_{ji}|$.

4.2 From SSC to Robust Subspace Clustering

The main issue with SSC (as described in Section 4.1) is that we only have access to the noisy data \mathbf{Y} ; that is, we do not see the matrix \mathbf{X} of covariates but rather a corrupted version \mathbf{Y} . This makes the problem challenging, as unlike conventional sparse recovery problems where only the response vector \mathbf{x}_i is corrupted, here both the covariates (columns of \mathbf{X}) and the response vector are corrupted. In particular, it may not be advisable to use (4.1.1) with \mathbf{y}_i and \mathbf{Y} in place of \mathbf{x}_i and \mathbf{X} as, strictly speaking, sparse representations no longer exist. Observe that the expression $\mathbf{x}_i = \mathbf{X}\boldsymbol{\beta}$ can be rewritten as $\mathbf{y}_i = \mathbf{Y}\boldsymbol{\beta} + (\mathbf{z}_i - \mathbf{Z}\boldsymbol{\beta})$. Viewing $(\mathbf{z}_i - \mathbf{Z}\boldsymbol{\beta})$ as a perturbation, it is natural to use ideas from sparse regression to obtain an estimate $\hat{\boldsymbol{\beta}}$, which is then used to construct the similarity matrix. In this thesis, we follow the same three-step procedure and shall focus on the first step; that is, on the construction of reliable similarity measures between pairs of points. Since we have corrupted data, we shall not use (4.1.1) here. For each form of corruption (noise, missing data, outliers, etc.) we shall give alternatives to (4.1.1) that are suited to suppressing that form of corruption. Also, we add denoising to Step III. The main steps in our Robust

Subspace Clustering (RSC) approach are shown in Algorithm 4. We shall use the acronym RSC-x for the different variations of the RSC scheme, where x is either N, M, O, D, or S representing Noise, Missing, Outliers, Dantzig formulation with noise, and sparse corruptions, respectively. For example, RSC-M refers to a variation of RSC that can handle missing data (detailed in Section 4.6).

Algorithm 4 Robust Subspace Clustering (RSC) procedure

Input: A data set \mathcal{Y} arranged as columns of $\mathbf{Y} \in \mathbb{R}^{n \times N}$.

1. For each $i \in \{1, \dots, N\}$, produce a sparse coefficient sequence $\{\hat{\beta}_i\}$ by regressing the i th vector \mathbf{y}_i onto the other columns of \mathbf{Y} . Collect these as columns of a matrix \mathbf{B} .
2. Form the similarity graph \mathcal{G} with nodes representing the N data points and edge weights given by $W_{ij} = |B_{ij}| + |B_{ji}|$.
3. Sort the eigenvalues $\delta_1 \geq \delta_2 \geq \dots \geq \delta_N$ of the normalized Laplacian of \mathcal{G} in descending order, and set

$$\hat{L} = N - \arg \max_{i=1, \dots, N-1} (\delta_i - \delta_{i+1}).$$

4. Apply a spectral clustering technique to the similarity graph using \hat{L} as the estimated number of clusters to obtain the partition $\mathcal{Y}_1, \dots, \mathcal{Y}_{\hat{L}}$.
5. Use PCA (or other robust subspace identification schemes) to find the best subspace fits ($\{S_\ell\}_1^L$) to each of the partitions ($\{\mathcal{Y}_\ell\}_1^L$) and denoise \mathbf{Y} as to obtain clean data points $\hat{\mathbf{X}}$.

Output: Subspaces $\{S_\ell\}_1^L$ and cleaned data points $\hat{\mathbf{X}}$.

4.2.1 Affine subspace clustering

RSC as discussed in Algorithm 4 clusters linear subspaces but can also cluster affine subspaces by adding the constraint $\mathbf{B}^T \mathbf{1} = \mathbf{1}$ to the regression problem in step 1. The idea behind this constraint is that when the subspaces are affine, while a point can no longer be written as a linear combination of points in the same subspace it can still be written as an affine combination of other points.

4.3 Performance metrics for similarity measures

Given the general structure of the method, we are interested in sparse regression techniques, which tend to select points in the same clusters (share the same underlying subspace) over those that do not share this property. Expressed differently, the hope is that whenever $B_{ij} \neq 0$, \mathbf{y}_i and \mathbf{y}_j originate from the same subspace. We introduce metrics to quantify performance.

Definition 4.3.1 (False discoveries) *Fix i and $j \in \{1, \dots, N\}$ and let \mathbf{B} be the outcome of Step 1 in Algorithm 4. Then we say that (i, j) obeying $B_{ij} \neq 0$ is a false discovery if \mathbf{y}_i and \mathbf{y}_j do not originate from the same subspace.*

Definition 4.3.2 (True discoveries) *In the same situation, (i, j) obeying $B_{ij} \neq 0$ is a true discovery if \mathbf{y}_j and \mathbf{y}_i originate from the same cluster/subspace.*

When there are no false discoveries, we shall say that the *subspace detection property* holds. In this case, the matrix \mathbf{B} is block diagonal after applying a permutation which makes sure that columns in the same subspace are contiguous. In some cases, the sparse regression method may select vectors from other subspaces and this property will not hold. However, it might still be possible to detect and construct reliable clusters by applying steps 2–5 in Algorithm 4.

4.4 Noisy data

This section introduces our methodology for subspace clustering when the data points are corrupted with noise. More specifically, in this section we assume that each point $\mathbf{y} \in \mathcal{Y}$ is of the form

$$\mathbf{y} = \mathbf{x} + \mathbf{z}, \quad (4.4.1)$$

where \mathbf{x} belongs to one of the subspaces and \mathbf{z} is a noise term. We suppose that the inverse signal-to-noise ratio (SNR) defined as $\|\mathbf{z}\|_{\ell_2}^2 / \|\mathbf{x}\|_{\ell_2}^2$ is bounded above. More

specifically, we assume that

$$\frac{\|\mathbf{z}\|_{\ell_2}}{\|\mathbf{x}\|_{\ell_2}} \leq \sigma^*,$$

where $\sigma^* < 1$ is a sufficiently small constant. From now on, we arrange the N observed data points as columns of a matrix $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_N] \in \mathbb{R}^{n \times N}$. With obvious notation, $\mathbf{Y} = \mathbf{X} + \mathbf{Z}$.

In practice, one may want to normalize the columns of the data matrix so that for all i , $\|\mathbf{y}_i\|_{\ell_2} = 1$ (R-code snippet for renormalizing a data point y is: `y <- y/sqrt(sum(y^2))`). Since with our SNR assumption, we have $\|\mathbf{y}\|_{\ell_2} \approx \|\mathbf{x}\|_{\ell_2} \sqrt{1 + \|\mathbf{z}\|_{\ell_2}^2}$ before normalization, then after normalization:

$$\mathbf{y} \approx \frac{1}{\sqrt{1 + \|\mathbf{z}\|_{\ell_2}^2}} (\mathbf{x} + \mathbf{z}),$$

where \mathbf{x} is unit-normed.

For ease of presentation, we work—in this section and in the proofs—with a model $\mathbf{y} = \mathbf{x} + \mathbf{z}$ in which $\|\mathbf{x}\|_{\ell_2} = 1$ instead of $\|\mathbf{y}\|_{\ell_2} = 1$ (the numerical Section 6.3 being the exception). The normalized model with $\|\mathbf{x}\|_{\ell_2} = 1$ is nearly the same as before. In particular, all of our methods and theoretical results in Section 5.4 hold with both models in which either $\|\mathbf{x}\|_{\ell_2} = 1$ or $\|\mathbf{y}\|_{\ell_2} = 1$.

We shall propose two different regression schemes to handle noise:

- (1) LASSO with data-driven regularization,
- (2) Bias-corrected Dantzig selector.

The two schemes are the subject of the next two sections.

4.4.1 LASSO with data-driven regularization

A natural sparse regression strategy is the LASSO:

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^N} \frac{1}{2} \|\mathbf{y}_i - \mathbf{Y}\boldsymbol{\beta}\|_{\ell_2}^2 + \lambda \|\boldsymbol{\beta}\|_{\ell_1} \quad \text{subject to } \beta_i = 0. \quad (4.4.2)$$

Whether such a methodology should succeed is unclear as we are not under a traditional model for both the response \mathbf{y}_i and the covariates \mathbf{Y} are noisy, see [?] for a discussion of sparse regression under matrix uncertainty and what can go wrong. We shall show later on that if one selects λ in a data-driven fashion, then compelling practical and theoretical performance can be achieved.

4.4.1.1 About as many true discoveries as dimension?

The nature of the problem is such that we wish to make few false discoveries (and not link too many pairs belonging to different subspaces) and so we would like to choose λ large. At the same time, we wish to make many true discoveries, whence a natural trade off. The reason why we need many true discoveries is that spectral clustering needs to assign points to the same cluster when they indeed lie near the same subspace. If the matrix \mathbf{B} is too sparse, this will not happen.

We now introduce a principle for selecting the regularization parameter; our exposition here is informal and we refer to Sections 5.4 and 7.3 for precise statements and proofs. Suppose we have noiseless data so that $\mathbf{Y} = \mathbf{X}$, and thus solve (4.1.1) with equality constraints. Under our model, assuming there are no false discoveries, the optimal solution is guaranteed to have exactly d —the dimension of the subspace the sample under study belongs to—nonzero coefficients with probability one. That is to say, when the point lies in a d -dimensional space, we find d ‘neighbors’.

The selection rule we shall analyze in this dissertation is to take λ as large as possible (as to prevent false discoveries) while making sure that the number of true discoveries is also on the order of the dimension d , typically in the range $[0.5d, 0.8d]$. We can say this differently. Imagine that all the points lie in the same subspace of dimension d so that every discovery is true. Then we wish to select λ in such a way that the number of discoveries is a significant fraction of d , the number one would get with noiseless data. Which value of λ achieves this goal? We will see in Section 4.4.1.2 that the answer is around $1/\sqrt{d}$. To put this in context, this means that we wish to select a regularization parameter which depends upon the dimension d of the subspace our point comes from. (We are aware that the dependence on d is unusual as in sparse regression the regularization parameter usually does not depend upon

the sparsity of the solution.) In turn, this immediately raises another question: since d is unknown, how can we proceed? In Section 4.4.1.4 we will see that it is possible to guess the dimension and construct fairly reliable estimates.

4.4.1.2 Data-dependent regularization

We now discuss values of λ obeying the demands formulated in the previous section. Our arguments are informal and we refer the reader to Section 5.4 for rigorous statements and to Section 7.3 for proofs. First, it simplifies the discussion to assume that we have no noise (the noisy case assuming $\sigma \ll 1$ is similar). Following our earlier discussion, imagine we have a vector $\mathbf{x} \in \mathbb{R}^n$ lying in the d -dimensional span of the columns of an $n \times N$ matrix \mathbf{X} . We are interested in values of λ so that the minimizer $\hat{\boldsymbol{\beta}}$ of the LASSO functional

$$K(\boldsymbol{\beta}, \lambda) = \frac{1}{2} \|\mathbf{x} - \mathbf{X}\boldsymbol{\beta}\|_{\ell_2}^2 + \lambda \|\boldsymbol{\beta}\|_{\ell_1}$$

has a number of nonzero components in the range $[0.5d, 0.8d]$, say. Now let $\hat{\boldsymbol{\beta}}_{\text{eq}}$ be the solution of the problem with equality constraints, or equivalently of the problem above with $\lambda \rightarrow 0^+$. Then

$$\frac{1}{2} \|\mathbf{x} - \mathbf{X}\hat{\boldsymbol{\beta}}\|_{\ell_2}^2 \leq K(\hat{\boldsymbol{\beta}}, \lambda) \leq K(\hat{\boldsymbol{\beta}}_{\text{eq}}, \lambda) = \lambda \|\hat{\boldsymbol{\beta}}_{\text{eq}}\|_{\ell_1}. \quad (4.4.3)$$

We make two observations: the first is that if $\hat{\boldsymbol{\beta}}$ has a number of nonzero components in the range $[0.5d, 0.8d]$, then $\|\mathbf{x} - \mathbf{X}\hat{\boldsymbol{\beta}}\|_{\ell_2}^2$ has to be greater than or equal to a fixed numerical constant. The reason is that we cannot approximate to arbitrary accuracy a generic vector living in a d -dimensional subspace as a linear combination of about $d/2$ elements from that subspace. The second observation is that $\|\hat{\boldsymbol{\beta}}_{\text{eq}}\|_{\ell_1}$ is on the order of \sqrt{d} , which is a fairly intuitive scaling (we have d coordinates, each of size about $1/\sqrt{d}$). This holds with the proviso that the algorithm operates correctly in the noiseless setting and does not select columns from other subspaces. Then (4.4.3) implies that λ has to scale at least like $1/\sqrt{d}$. On the other hand, $\hat{\boldsymbol{\beta}} = \mathbf{0}$ if $\lambda \geq \|\mathbf{X}^T \mathbf{x}\|_{\ell_\infty}$. Now the informed reader knows that $\|\mathbf{X}^T \mathbf{x}\|_{\ell_\infty}$ scales at most like

$\sqrt{(\log N)/d}$ so that choosing λ around this value yields no discovery (one can refine this argument to show that λ cannot be higher than a constant times $1/\sqrt{d}$ as we would otherwise have a solution that is too sparse). Hence, λ is around $1/\sqrt{d}$.

It might be possible to compute a precise relationship between λ and the expected number of true discoveries in an asymptotic regime in which the number of points and the dimension of the subspace both increase to infinity in a fixed ratio by adapting ideas from [31, 33]. We will not do so here as this is beyond the scope of this paper. Rather, we investigate this relationship by means of a numerical study.

Here, we fix a single subspace in \mathbb{R}^n with $n = 2,000$. We use a sampling density equal to $\rho = 5$ and vary the dimension $d \in \{10, 20, 50, 100, 150, 200\}$ of the subspace as well as the noise level $\sigma \in \{0.25, 0.5\}$. For each data point, we solve (4.4.2) for different values of λ around the heuristic $\lambda_o = 1/\sqrt{d}$, namely, $\lambda \in [0.1\lambda_o, 2\lambda_o]$. In our experiments, we declare a discovery if an entry in the optimal solution exceeds 10^{-3} . Figures 4.1a and 4.1b show the number of discoveries per subspace dimension (the number of discoveries divided by d). One can clearly see that the curves corresponding to various subspace dimensions stack up on top of each other, thereby confirming that a value of λ on the order of $1/\sqrt{d}$ yields a fixed fraction of true discoveries. Further inspection also reveals that the fraction of true discoveries is around 50% near $\lambda = \lambda_o$, and around 75% near $\lambda = \lambda_o/2$. We have observed empirically that increasing ρ typically yields a slight increase in the fraction of true discoveries (unless, of course, ρ is exponentially large in d).

4.4.1.3 The false-true discovery trade off

We now show empirically that in our model choosing λ around $1/\sqrt{d}$ typically yields very few false discoveries as well as many true discoveries; this holds with the proviso that the subspaces are of course not very close to each other.

In this simulation, 22 subspaces of varying dimensions in \mathbb{R}^n with $n = 2,000$ have been independently selected uniformly at random; there are 5, 4, 3, 4, 4, and 2 subspaces of respective dimensions 200, 150, 100, 50, 20 and 10. This is a challenging regime since the sum of the subspace dimensions equals 2,200 and exceeds the ambient dimension (the clean data matrix \mathbf{X} has full rank). We use a sampling density equal to

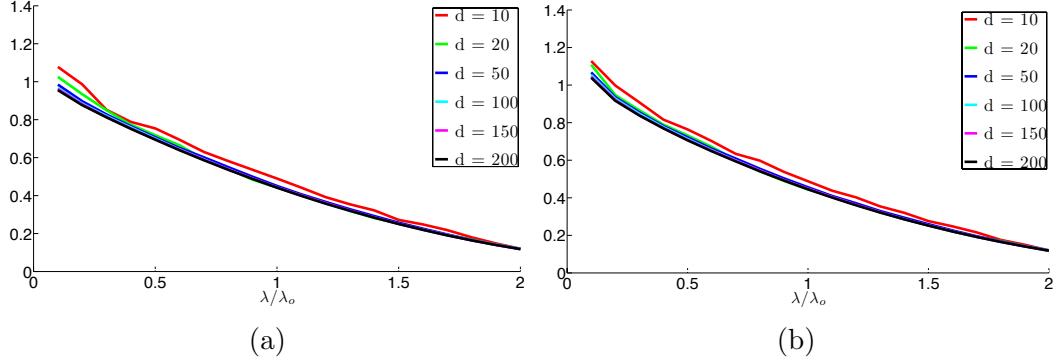


Figure 4.1: Average number of true discoveries normalized by subspace dimension for values of λ in an interval including the heuristic $\lambda_o = 1/\sqrt{d}$. (a) $\sigma = 0.25$. (b) $\sigma = 0.5$.

$\rho = 5$ for each subspace and set the noise level to $\sigma = 0.3$. To evaluate the performance of the optimization problem (4.4.2), we proceed by selecting a subset of columns as follows: for each dimension, we take 100 cases at random belonging to subspaces of that dimension. Hence, the total number of test cases is $m = 600$ so that we only solve m optimization problems (4.4.2) out of the total N possible cases. Below, $\beta^{(i)}$ is the solution to (4.4.2) and $\beta_S^{(i)}$ its restriction to columns with indices in the same subspace. Hence, a nonvanishing entry in $\beta_S^{(i)}$ is a true discovery and, likewise, a nonvanishing entry in $\beta_{S^c}^{(i)}$ is false. For each data point we sweep the tuning parameter λ in (4.4.2) around the heuristic $\lambda_o = 1/\sqrt{d}$ and work with $\lambda \in [0.05\lambda_o, 2.5\lambda_o]$. In our experiments, a discovery is a value obeying $|B_{ij}| > 10^{-3}$.

In analogy with the signal detection literature we view the empirical averages of $\|\beta_{S^c}^{(i)}\|_{\ell_0}/(n - d)$ and $\|\beta_S^{(i)}\|_{\ell_0}/d$ as False Positive Rate (FPR) and True Positive Rate (TPR). On the one hand, Figures 4.2a and 4.2b show that for values around $\lambda = \lambda_o$, the FPR is zero (so there are no false discoveries). On the other hand, Figure 4.2c shows that the TPR curves corresponding to different dimensions are very close to each other and resemble those in Figure 14.4 in which all the points belong to the same cluster with no opportunity of making a false discovery. Hence, taking λ near $1/\sqrt{d}$ gives a performance close to what can be achieved in a noiseless situation. That is to say, we have no false discovery and a number of true discoveries about $d/2$ if we choose $\lambda = \lambda_o$. Figure 4.2d plots TPR versus FPR (a.k.a. the Receiver Operating Characteristic (ROC) curve) and indicates that $\lambda = \lambda_o$ (marked by a red dot) is

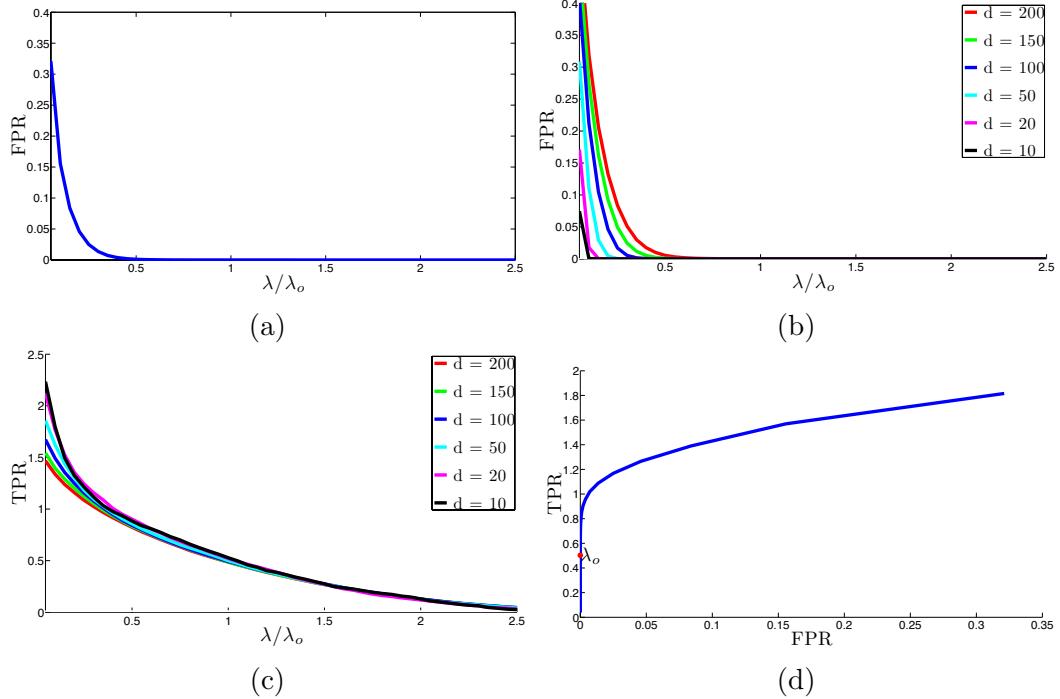


Figure 4.2: Performance of LASSO for values of λ in an interval including the heuristic $\lambda_o = 1/\sqrt{d}$. (a) Average number of false discoveries normalized by $(n - d)$ (FPR) on all m sampled data points. (b) FPR for different subspace dimensions. Each curve represents the average FPR over those samples originating from subspaces of the same dimension. (c) Average number of true discoveries per dimension for various dimensions (TPR). (d) TPR vs. FPR (ROC curve). The point corresponding to $\lambda = \lambda_o$ is marked as a red dot.

an attractive trade-off as it provides no false discoveries and sufficiently many true discoveries.

4.4.1.4 A two-step procedure

Returning to the selection of the regularization parameter, we would like to use λ on the order of $1/\sqrt{d}$. However, we do not know d and proceed by substituting an estimate. In the next section, we will see that we are able to quantify theoretically the performance of the following proposal: (1) run a hard constrained version of the LASSO and use an estimate \hat{d} of dimension based on the ℓ_1 norm of the fitted coefficient sequence; (2) impute a value for λ constructed from \hat{d} . The two-step

procedure is explained in Algorithm 5. We shall refer to this approach as RSC-N. Again, our exposition is informal here and we refer to Section 7.3 for precise statements.

Algorithm 5 Two-step procedure with data-driven regularization (RSC-N)

for $i = 1, \dots, N$ **do**

 1. Solve

$$\boldsymbol{\beta}^* = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^N} \|\boldsymbol{\beta}\|_{\ell_1} \quad \text{subject to} \quad \|\mathbf{y}_i - \mathbf{Y}\boldsymbol{\beta}\|_{\ell_2} \leq \tau \quad \text{and} \quad \boldsymbol{\beta}_i = 0. \quad (4.4.4)$$

 2. Set $\lambda = f(\|\boldsymbol{\beta}^*\|_{\ell_1})$.

 3. Solve

$$\hat{\boldsymbol{\beta}} = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^N} \frac{1}{2} \|\mathbf{y}_i - \mathbf{Y}\boldsymbol{\beta}\|_{\ell_2}^2 + \lambda \|\boldsymbol{\beta}\|_{\ell_1} \quad \text{subject to} \quad \boldsymbol{\beta}_i = 0.$$

 4. Set $\mathbf{B}_i = \hat{\boldsymbol{\beta}}$.

end for

To understand the rationale behind this, imagine we have noiseless data—i. e. $\mathbf{Y} = \mathbf{X}$ —and are solving (4.1.1), which simply is our first step (4.4.4) with the proviso that $\tau = 0$. When there are no false discoveries, one can show that the ℓ_1 norm of $\boldsymbol{\beta}^*$ is roughly of size \sqrt{d} as shown in Lemma 7.3.2 from Section 7.3. This suggests using a multiple of $\|\boldsymbol{\beta}^*\|_{\ell_1}$ as a proxy for \sqrt{d} . To drive this point home, take a look at Figure 4.3a which solves (4.4.4) with the same data as in the previous example and $\tau = 2\sigma$. The plot reveals that the values of $\|\boldsymbol{\beta}^*\|_{\ell_1}$ fluctuate around \sqrt{d} . This is shown more clearly in Figure 4.3b, which shows that $\|\boldsymbol{\beta}^*\|_{\ell_1}$ is concentrated around $\frac{1}{4}\sqrt{d}$ with, as expected, higher volatility at lower values of dimension.

Under suitable assumptions, we shall see in Section 5.4 that with noisy data, there are simple rules for selecting τ that guarantee, with high probability, that there are no false discoveries. To be concrete, one can take $\tau = 2\sigma$ and $f(t) \propto t^{-1}$. Returning to our running example, we have $\|\boldsymbol{\beta}^*\|_{\ell_1} \approx \frac{1}{4}\sqrt{d}$. Plugging this into $\lambda = 1/\sqrt{d}$ suggests taking $f(t) \approx 0.25t^{-1}$. The plots in Figure 4.4 demonstrate that this is indeed effective. Experiments in Section 6.3 indicate that this is a good choice on real data as well.

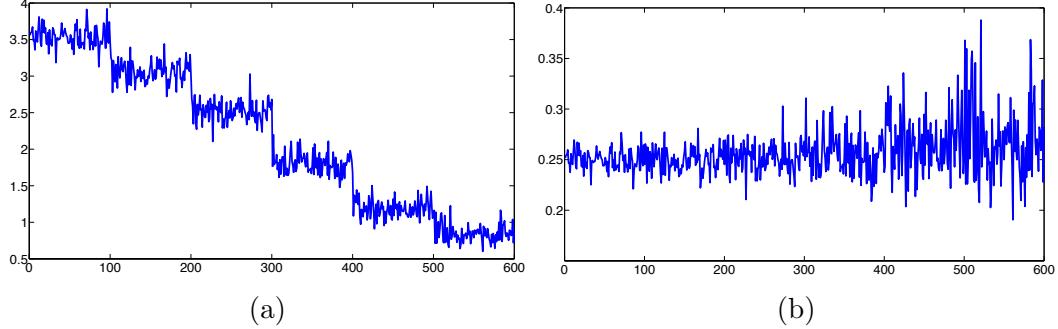


Figure 4.3: Optimal values of (4.4.4) for 600 samples using $\tau = 2\sigma$. The first 100 values correspond to points originating from subspaces of dimension $d = 200$, the next 100 from those of dimension $d = 150$, and so on through $d \in \{100, 50, 20, 10\}$. (a) Value of $\|\boldsymbol{\beta}^*\|_{\ell_1}$. (b) Value of $\|\boldsymbol{\beta}^*\|_{\ell_1}/\sqrt{d}$.

The two-step procedure requires solving two LASSO problems for each data point and is useful when there are subspaces of large dimensions (in the hundreds, say) and some others of low-dimensions (three or four, say). In some applications such as motion segmentation in computer vision, the dimensions of the subspaces are all equal and known in advance [224]. In this case, one can forgo the two-step procedure and simply set $\lambda = 1/\sqrt{d}$.

4.4.2 The Bias-corrected Dantzig Selector

One can think of other ways of performing the first step in Algorithm 4 in the presence of noisy data and this section discusses another approach based on a modification of the Dantzig selector, a popular sparse regression technique [66]. Unlike the two-step procedure, we do not claim any theoretical guarantees for this method and shall only explore its properties on real and simulated data.

Applied directly to our problem, the Dantzig selector takes the form

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^N} \|\boldsymbol{\beta}\|_{\ell_1} \quad \text{subject to} \quad \|\mathbf{Y}_{(-i)}^T(\mathbf{y}_i - \mathbf{Y}\boldsymbol{\beta})\|_{\ell_\infty} \leq \lambda \quad \text{and} \quad \boldsymbol{\beta}_i = 0, \quad (4.4.5)$$

where $\mathbf{Y}_{(-i)}$ is \mathbf{Y} with the i th column deleted. However, this is hardly suitable since the design matrix \mathbf{Y} is corrupted. Interestingly, recent work [?, ?] has studied the

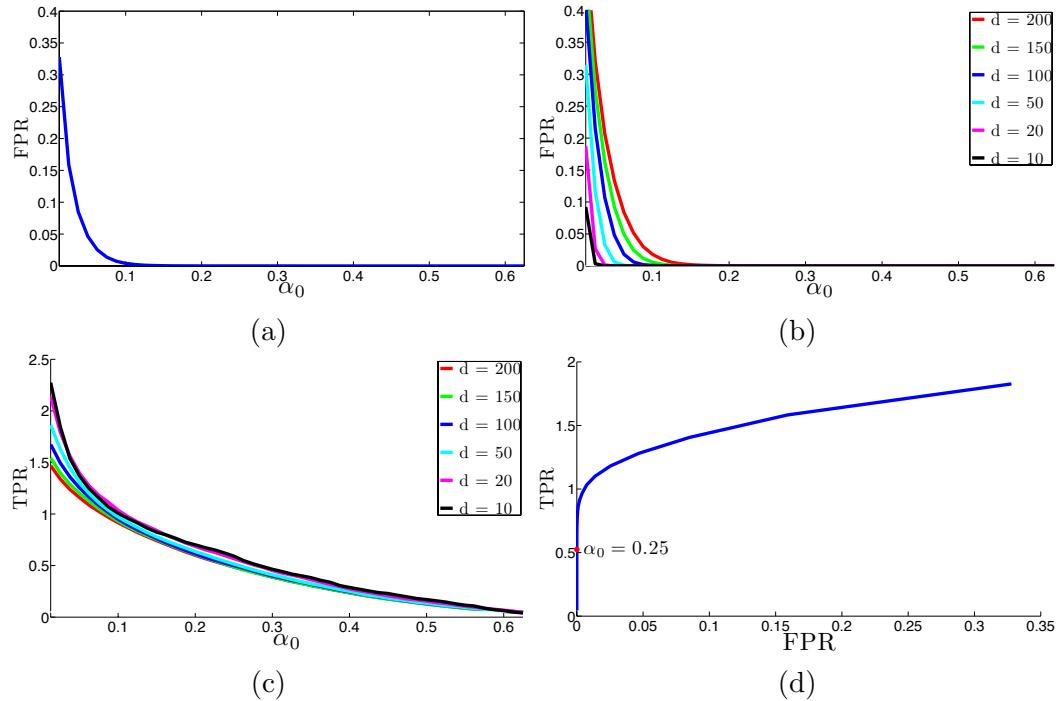


Figure 4.4: Performance of the two-step procedure using $\tau = 2\sigma$ and $f(t) = \alpha_0 t^{-1}$ for values of α_0 around the heuristic $\alpha_0 = 0.25$. (a) False positive rate (FPR). (b) FPR for various subspace dimensions. (c) True positive rate (TPR). (d) TPR vs. FPR.

problem of estimating a sparse vector from the standard linear model under uncertainty in the design matrix. The setup in these papers is close to our problem and we propose a modified Dantzig selection procedure inspired but not identical to the methods set forth in [?, ?].

4.4.2.1 The correction

If we had clean data, we would solve (4.1.1); this is (4.4.5) with $\mathbf{Y} = \mathbf{X}$ and $\lambda = 0$. Let $\boldsymbol{\beta}^I$ be the solution to this ideal noiseless problem. Applied to our problem, the main idea in [?, ?] would be to find a formulation that resembles (4.4.5) with the property that $\boldsymbol{\beta}^I$ is feasible. Since $\mathbf{x}_i = \mathbf{X}_{(-i)}\boldsymbol{\beta}^I_{(-i)}$, observe that we have the following decomposition:

$$\begin{aligned}\mathbf{Y}_{(-i)}^T(\mathbf{y}_i - \mathbf{Y}\boldsymbol{\beta}^I) &= (\mathbf{X}_{(-i)} + \mathbf{Z}_{(-i)})^T(\mathbf{z}_i - \mathbf{Z}\boldsymbol{\beta}^I) \\ &= \mathbf{X}_{(-i)}^T(\mathbf{z}_i - \mathbf{Z}\boldsymbol{\beta}^I) + \mathbf{Z}_{(-i)}^T\mathbf{z}_i - \mathbf{Z}_{(-i)}^T\mathbf{Z}\boldsymbol{\beta}^I.\end{aligned}$$

Then the conditional mean is given by

$$\mathbb{E}[\mathbf{Y}_{(-i)}^T(\mathbf{y}_i - \mathbf{Y}\boldsymbol{\beta}^I) | \mathbf{X}] = -\mathbb{E}\mathbf{Z}_{(-i)}^T\mathbf{Z}\boldsymbol{\beta}^I_{(-i)} = -\sigma^2\boldsymbol{\beta}^I_{(-i)}.$$

In other words,

$$\sigma^2\boldsymbol{\beta}^I_{(-i)} + \mathbf{Y}_{(-i)}^T(\mathbf{y}_i - \mathbf{Y}\boldsymbol{\beta}^I) = \boldsymbol{\xi}$$

where $\boldsymbol{\xi}$ has mean zero. In Section 4.4.2.2, we compute the variance of the j th component ξ_j , given by

$$\mathbb{E}\xi_j^2 = \frac{\sigma^2}{n}(1 + \|\boldsymbol{\beta}^I\|_{\ell_2}^2) + \frac{\sigma^4}{n}(1 + (\beta_j^I)^2 + \|\boldsymbol{\beta}^I\|_{\ell_2}^2). \quad (4.4.6)$$

Owing to our Gaussian assumptions, $|\xi_j|$ shall be smaller than 3 or 4 times this standard deviation, say, with high probability.

Hence, we may want to consider a procedure of the form

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^N} \|\boldsymbol{\beta}\|_{\ell_1} \quad \text{subject to} \quad \|\mathbf{Y}_{(-i)}^T(\mathbf{y}_i - \mathbf{Y}\boldsymbol{\beta}) + \sigma^2\boldsymbol{\beta}_{(-i)}\|_{\ell_\infty} \leq \lambda \quad \text{and} \quad \boldsymbol{\beta}_i = 0. \quad (4.4.7)$$

We shall refer to this approach as *RSC – D*. It follows that if we take λ to be a reasonable multiple of (4.4.6), then β^I would obey the constraint in (4.4.7) with high probability. Hence, we would need to approximate the variance (4.4.6). Numerical simulations together with asymptotic calculations presented in Appendix D give that $\|\beta^I\|_{\ell_2} \leq 1$ with very high probability. Thus neglecting the term in $(\beta_j^I)^2$,

$$\mathbb{E} \xi_j^2 \approx \frac{\sigma^2}{n} (1 + \sigma^2) (1 + \|\beta^I\|_{\ell_2}^2) \leq 2 \frac{\sigma^2}{n} (1 + \sigma^2).$$

This suggests taking λ to be a multiple of $\sqrt{2/n} \sigma \sqrt{1 + \sigma^2}$. This is interesting because the parameter λ does not depend on the dimension of the underlying subspace. We shall refer to (4.4.7) as the *bias-corrected Dantzig selector*, which resembles the proposal in [?, ?] for which the constraint is a bit more complicated and of the form $\|\mathbf{Y}_{(-i)}^T (\mathbf{y}_i - \mathbf{Y}\beta) + \mathbf{D}_{(-i)}\beta\|_{\ell_\infty} \leq \mu \|\beta\|_{\ell_1} + \lambda$.

To get a sense about the validity of this proposal, we test it on our running example by varying $\lambda \in [\lambda_o, 8\lambda_o]$ around the heuristic $\lambda_o = \sqrt{2/n} \sigma \sqrt{1 + \sigma^2}$. Figure 4.5 shows that good results are achieved around factors in the range [4, 6].

In our synthetic simulations, both the two-step procedure and the corrected Dantzig selector seem to be working well in the sense that they yield many true discoveries while making very few false discoveries, if any. Comparing Figures 4.5b and 4.5c with those from Section 4.4.1 show that the corrected Dantzig selector has more true discoveries for subspaces of small dimensions (they are essentially the same for subspaces of large dimensions); that is, the two-step procedure is more conservative when it comes to subspaces of smaller dimensions. As explained earlier this is due to our conservative choice of λ resulting in a TPR about half of what is obtained in a noiseless setting. Having said this, it is important to keep in mind that in these simulations the planes are drawn at random and as a result, they are sort of far from each other. This is why a less conservative procedure can still achieve a low FPR. When subspaces of smaller dimensions are closer to each other or when the statistical model does not hold exactly as in real data scenarios, a conservative procedure may be more effective. In fact, experiments on real data in Section 6.3 confirm this and show that for the corrected Dantzig selector, one needs to choose values much larger

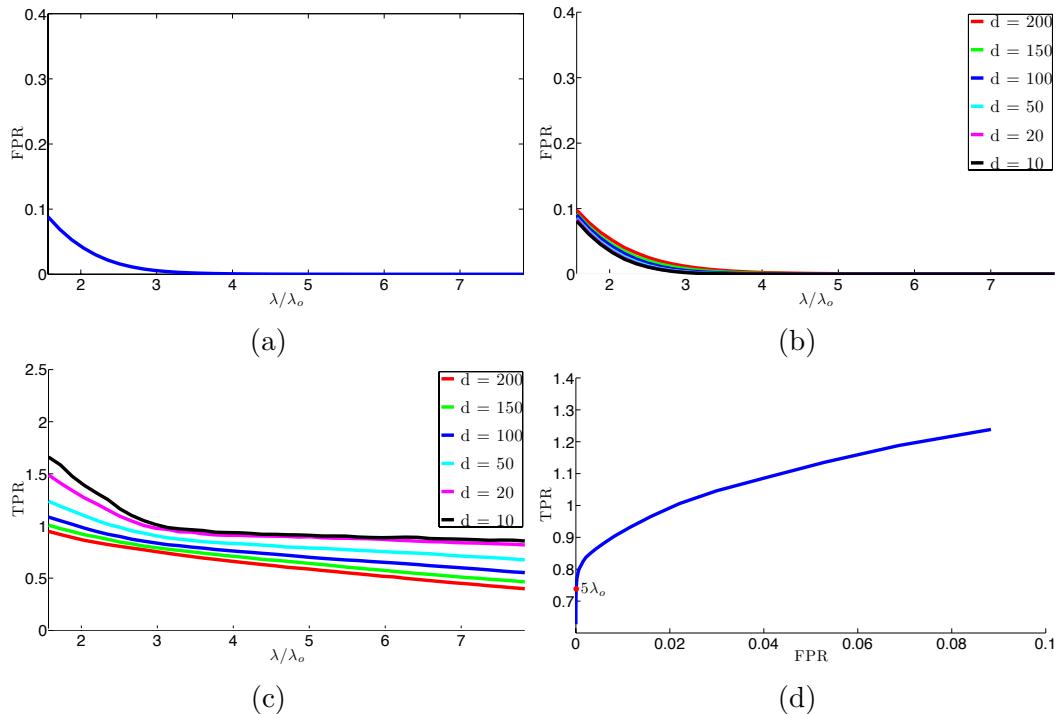


Figure 4.5: Performance of the bias-corrected Dantzig selector for values of λ that are multiples of the heuristic $\lambda_o = \sqrt{2/n} \sigma \sqrt{1 + \sigma^2}$. (a) False positive rate (FPR). (b) FPR for different subspace dimensions. (c) True positive rate (TPR). (d) TPR vs. FPR.

than λ_o to yield good results.

4.4.2.2 Variance calculation

By definition,

$$\xi_j = \langle \mathbf{x}_j, \mathbf{z}_i - \mathbf{Z}\boldsymbol{\beta}^I \rangle + \langle \mathbf{z}_j, \mathbf{z}_i \rangle - (\mathbf{z}_j^T \mathbf{z}_j - \sigma^2) \beta_j^I - \sum_{k:k \neq i,j} \mathbf{z}_j^T \mathbf{z}_k \beta_k^I := I_1 + I_2 + I_3 + I_4.$$

A simple calculation shows that for $\ell_1 \neq \ell_2$, $\text{Cov}(I_{\ell_1}, I_{\ell_2}) = 0$ so that

$$\mathbb{E} \xi_j^2 = \sum_{\ell=1}^4 \text{Var}(I_\ell).$$

We compute

$$\begin{aligned} \text{Var}(I_1) &= \frac{\sigma^2}{n} (1 + \|\boldsymbol{\beta}^I\|_{\ell_2}^2), & \text{Var}(I_3) &= \frac{\sigma^4}{n} 2(\beta_j^I)^2, \\ \text{Var}(I_2) &= \frac{\sigma^4}{n}, & \text{Var}(I_4) &= \frac{\sigma^4}{n} [\|\boldsymbol{\beta}^I\|_{\ell_2}^2 - (\beta_j^I)^2] \end{aligned}$$

and (4.4.6) follows.

4.5 Gross outliers

We now turn our attention to the case where the point set is corrupted in the sense that there are N_0 outliers assumed to be distributed uniformly at random on the unit sphere. Here, we wish to correctly identify the outlier points and apply any of the subspace clustering algorithms to the remaining samples. We propose a very simple detection procedure for this task. As in SSC, decompose each \mathbf{x}_i as a linear combination of all the other points by solving an ℓ_1 -minimization problem. Then one expects the expansion of an outlier to be less sparse. This suggests the following detection rule: declare \mathbf{x}_i to be an outlier if and only if the optimal value of (4.1.1) is above a fixed threshold. This makes sense because if \mathbf{x}_i is an outlier, one expects the optimal value to be on the order of \sqrt{n} (provided N is at most polynomial in n)

whereas this value will be at most on the order of \sqrt{d} if \mathbf{x}_i belongs to a subspace of dimension d . In short, we expect a gap—a fact we will make rigorous in the next chapter. The main steps of the procedure are shown in Algorithm 6. We shall refer to this approach as RSC-O.

Algorithm 6 Subspace clustering in the presence of outliers (RSC-O)

Input: A data set \mathcal{X} arranged as columns of $\mathbf{X} \in \mathbb{R}^{n \times N}$.

1. Solve

$$\begin{aligned} & \text{minimize} && \|\mathbf{B}\|_{\ell_1} \\ & \text{subject to} && \mathbf{X}\mathbf{B} = \mathbf{X} \\ & && \text{diag}(\mathbf{B}) = \mathbf{0}. \end{aligned}$$

2. For each $i \in \{1, \dots, N\}$, declare i to be an outlier iff $\|\beta_i\|_{\ell_1} > \kappa(\gamma)\sqrt{n}$.²

3. Apply a subspace clustering technique to the remaining points.

Output: Partition $\mathcal{X}_0, \mathcal{X}_1, \dots, \mathcal{X}_L$.

4.6 Missing data

Here we explain our method for subspace clustering when some entries of the data matrix are missing. This problem may also be viewed as a generalization of standard low-rank matrix completion to cases where the matrix is of high or potentially full-rank. The goal is two fold:

- 1- partition the data into different clusters based on subspace of origin and approximate the underlying subspaces.
- 2- complete the missing entries.

Here we focus on step 1. Upon finding the correct clustering, one can apply any one of the low-rank matrix recovery algorithms on each cluster to complete the missing entries.

We begin with some notation. We will use $\bar{\delta}$ to denote the fraction of entries that are missing (number of missing entries divided by total number of entries). We also

²Here $\gamma = \frac{N-1}{n}$ and κ is a threshold ratio function whose value shall be discussed later on.

use $\mathbf{x}^{(i)}$ to denote the i th column of \mathbf{X} , $\Omega_i \subset \{1, 2, \dots, n\}$ to denote the observations we have from the i -th column of \mathbf{X} and \mathbf{X}_{Ω_i} to denote the submatrix of \mathbf{X} with rows selected by Ω_i . $\mathbf{x}_{\Omega_i}^{(i)}$ is defined in a similar manner.

To apply our RSC procedure we have to come up with a good regression technique that works well with missing data.³ If there were no missing entries in the covariates of (4.1.1) we would solve the following sequence of idealized problems

$$\boldsymbol{\beta}^I(i) = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^N} \|\boldsymbol{\beta}\|_{\ell_1} \quad \text{subject to} \quad \mathbf{X}_{\Omega_i} \boldsymbol{\beta} = \mathbf{x}_{\Omega_i}^{(i)} \text{ and } \beta_i = 0. \quad (4.6.1)$$

which is problem (4.1.1) with $\lambda = 0$ and $\bar{\delta} = 0$ on the covariates.

Set $\boldsymbol{\gamma}_i$ equal to $\mathbf{X}_{\Omega_i}^T \mathbf{x}_{\Omega_i}^{(i)}$ with the i th entry set to zero. Also, set $\boldsymbol{\Gamma}_i$ equal to $\mathbf{X}_{\Omega_i}^T \mathbf{X}_{\Omega_i}$ with the i th row and column set to 0. Therefore, (4.6.1) can be rewritten in the form

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^N} \|\boldsymbol{\beta}\|_{\ell_1} \quad \text{subject to} \quad \boldsymbol{\gamma}_i - \boldsymbol{\Gamma}_i \boldsymbol{\beta} = 0 \text{ and } \beta_i = 0. \quad (4.6.2)$$

However, we do not get to see all of the entries of \mathbf{X}_{Ω_i} , and can not build the correct $\boldsymbol{\gamma}_i$ and $\boldsymbol{\Gamma}_i$. Therefore, we will try to estimate these quantities. For this purpose, we build a matrix \mathbf{Y} based on the observed entries of \mathbf{X} as follows

$$Y_{ij} = \begin{cases} \frac{X_{ij}}{(1-\delta)} & \text{if observed} \\ 0 & \text{if missing} \end{cases}. \quad (4.6.3)$$

Set $\widehat{\boldsymbol{\Gamma}}_i$ equal to $\mathbf{Y}_{\Omega_i}^T \mathbf{Y}_{\Omega_i} - \bar{\delta} \text{diag}(\mathbf{Y}_{\Omega_i}^T \mathbf{Y}_{\Omega_i})$ with the i th row and column set to zero. Similarly, set $\widehat{\boldsymbol{\gamma}}_i$ equal to $\mathbf{Y}_{\Omega_i}^T \mathbf{y}_{\Omega_i}^{(i)}$ with the i th row set to zero. One justification for this choice is that when the observed entries are revealed at random $(\widehat{\boldsymbol{\Gamma}}_i, \widehat{\boldsymbol{\gamma}}_i)$ is an unbiased estimator for $(\boldsymbol{\Gamma}_i, \boldsymbol{\gamma}_i)$. This motivates the following bias-corrected Dantzig selector

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^N} \|\boldsymbol{\beta}\|_{\ell_1} \quad \text{subject to} \quad \|\widehat{\boldsymbol{\gamma}}_i - \widehat{\boldsymbol{\Gamma}}_i \boldsymbol{\beta}\|_{\ell_\infty} \leq \lambda \text{ and } \beta_i = 0. \quad (4.6.4)$$

We note that this is a variation of the Bias-corrected Dantzig selector of Section 4.4.2 appropriately modified so as to handle missing data. We shall refer to this approach

³Please also see the papers [156, 208, 209] for related approaches to regression with missing data.

as RSC-M.

4.6.1 Detailed implementation

In this section we shall explain how the procedure proposed above for handling missing data is implemented on actual data. The casual reader may choose to skip this section on a first read. For the sake of simplicity in this section we focus on one optimization problem of the form (4.6.4). We assume that \mathbf{x} is a point from S_1 and the rest of the data points are arranged as columns of a matrix \mathbf{X} . We use Ω to denote the index of the entries revealed to us from point \mathbf{x} . Similarly, we use Ω_j to denote the index of the entries we get to observe from the j th column of \mathbf{X} . As we described earlier we set

$$\widehat{\boldsymbol{\Gamma}} = \mathbf{Y}_\Omega^T \mathbf{Y}_\Omega - \delta \text{diag}(\mathbf{Y}_\Omega^T \mathbf{Y}_\Omega), \quad \widehat{\boldsymbol{\gamma}} = \mathbf{Y}_\Omega^T \mathbf{x}_\Omega,$$

and solve problems of the form

$$\min \|\boldsymbol{\beta}\|_{\ell_1} \quad \text{subject to} \quad \|\widehat{\boldsymbol{\gamma}} - \widehat{\boldsymbol{\Gamma}}\boldsymbol{\beta}\|_{\ell_\infty} \leq \lambda. \quad (4.6.5)$$

We have made two idealized assumptions in the algorithm presented so far (1) we assumed that the data points (columns of \mathbf{X}) have unit Euclidean norm (2) the fraction of missing entries $\bar{\delta}$ is roughly the same for each column. In this section we explain the correct way to address these issues. We use $\hat{\boldsymbol{\delta}} \in \mathbb{R}^{N-1}$ to denote the empirical fraction of data missing from each column of \mathbf{X} . That is, $\hat{\delta}_j = 1 - |\Omega_j|/n$. As explained before, we use $\bar{\delta}$ to denote the empirical fraction of missing entries (number of missing entries divided by total number of entries). Similar to what we explained earlier we first assume that the missing entries are zeros and build the following matrix

$$\tilde{Y}_{ij} = \begin{cases} X_{ij} & \text{if observed} \\ 0 & \text{if missing} \end{cases}.$$

Next we normalize the columns of $\tilde{\mathbf{Y}}$ so that each column has unit Euclidean norm. Call the resulting matrix $\bar{\mathbf{Y}}$ and set $\mathbf{Y} = \bar{\mathbf{Y}} \text{diag}(\mathbf{w})$ where $\mathbf{w}_i = \frac{1}{\sqrt{1-\hat{\delta}_i}}$. Note that

in (4.6.3) before rescaling each column had Euclidean norm roughly equal to $\sqrt{1 - \bar{\delta}}$ here we use a rescaling by $1/\sqrt{1 - \bar{\delta}}$ instead of $1/(1 - \bar{\delta})$ since each column of $\bar{\mathbf{Y}}$ has unit Euclidean norm. In a similar manner we define $\tilde{\mathbf{y}}$ and normalize it to $\bar{\mathbf{y}}$. Now we define $\mathbf{y} = \sqrt{1 - \bar{\delta}}\tilde{\mathbf{y}}$ and set

$$\widehat{\Gamma} = \mathbf{Y}_\Omega^T \mathbf{Y}_\Omega - \text{diag}(\widehat{\boldsymbol{\delta}})\text{diag}(\mathbf{Y}_\Omega^T \mathbf{Y}_\Omega), \quad \widehat{\boldsymbol{\gamma}} = \mathbf{Y}_\Omega^T \mathbf{y}_\Omega, \quad \lambda = \frac{\sqrt{2 \log N}}{\sqrt{n}} \sqrt{\frac{\bar{\delta}}{1 - \bar{\delta}}}.$$

4.7 Sparse corruption

As we have seen so far one of the advantages of the sparse regression formulation for subspace clustering is that it can handle various forms of corruption. Another form of corruption that occurs naturally in data is sparse corruption. More specifically, each point $\mathbf{y} \in \mathcal{Y}$ is of the form

$$\mathbf{y} = \mathbf{x} + \mathbf{z},$$

where \mathbf{x} belongs to one of the subspaces and \mathbf{z} is a sparse vector with the non-zero entries representing the sparse corruptions. In this case a natural way to modify (4.1.1) is to use

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^N} \frac{1}{2} \|\mathbf{y}_i - \mathbf{Y}\boldsymbol{\beta}\|_{\ell_1} + \lambda \|\boldsymbol{\beta}\|_{\ell_1} \quad \text{subject to } \beta_i = 0.$$

This approach was suggested in [98]. We shall refer to this approach as RSC-S. Observe that the expression $\mathbf{x}_i = \mathbf{X}\boldsymbol{\beta}$ can be rewritten as $\mathbf{y}_i = \mathbf{Y}\boldsymbol{\beta} + (\mathbf{z}_i - \mathbf{Z}\boldsymbol{\beta})$. Since $(\mathbf{z}_i - \mathbf{Z}\boldsymbol{\beta})$ can be viewed as a sparse or heavy tailed perturbation, it is natural to impose the ℓ_1 penalty on $\mathbf{y}_i - \mathbf{Y}\boldsymbol{\beta}$.

Chapter 5

Theory

In this chapter we shall present our main theoretical results concerning SSC and its robust variations (different versions of RSC).

5.1 Modeling assumptions

Throughout this chapter we shall focus on *linear* subspace clustering. Our model assumes that each point $\mathbf{y} \in \mathcal{Y}_1 \cup \mathcal{Y}_2 \cup \dots \cup \mathcal{Y}_L$ is of the form

$$\mathbf{y} = \mathbf{x} + \mathbf{z}, \quad (5.1.1)$$

where \mathbf{x} denotes the “clean” data point which belongs to one of the subspaces and \mathbf{z} denotes the corruption on this point. For ease of presentation, throughout we will assume that the clean data point have unit norm i.e. $\|\mathbf{x}\|_{\ell_2} = 1$. As explained in Chapter 4 one can apply simple preprocessing steps to the observed data points $\mathbf{y} \in \mathcal{Y}$ such that this assumption is roughly true. As mentioned before we will gather all of the observed data points as columns of a matrix $\mathbf{Y} \in \mathbb{R}^{n \times N}$. and obvious notation $\mathbf{Y} = \mathbf{X} + \mathbf{Z}$.

5.1.1 Models for the clean data points

In order to better understand the regime in which different subspace clustering algorithms succeed as well as their limitations, we will consider three different models for the noiseless or “clean” data points. Our aim is to use these models to give informative bounds that highlight the dependence upon key parameters of the problem such as (1) the number of subspaces, (2) the dimensions of these subspaces, (3) the relative orientations of these subspaces, (4) the number of data points per subspace and so on.

- *Deterministic model.* In this model the orientation of the subspaces as well as the distribution of the points on each subspace are nonrandom.
- *Semi-random model.* Here, the subspaces are fixed but the points are distributed uniformly at random on the unit sphere of each of the subspaces.
- *Fully random model.* Here, both the orientation of the subspaces and the distribution of the points are random.

5.1.2 Models for corruption

In order to better understand when the different robust subspace clustering techniques presented in Chapter 4 are effective we will assume that the corruption vectors \mathbf{z} are stochastic terms. Below we explain our specific random models for each form of corruption:

- *Noisy data.* In this model we assume that the corruption vector \mathbf{z} on each of the data points are i.i.d. random vectors distributed as $\mathcal{N}(\mathbf{0}, \frac{\sigma^2}{n} \mathbf{I})$. Each observation is thus the superposition of a noiseless sample taken from one of the subspaces and of a stochastic perturbation whose Euclidean norm is about σ times the signal strength so that $\mathbb{E} \|\mathbf{z}\|_{\ell_2}^2 = \sigma^2 \|x\|_{\ell_2}^2$.
- *Missing data.* In this model instead of observing the clean data matrix \mathbf{X} , a small subset of the entries of such a matrix is revealed. We assume a missing at

random model were each entry is revealed independently with probability $1 - \delta$. As explained previously, by filling the missing entries with zero, this form of corruption can also be viewed as an additive corruption.

- *Gross outliers.* In this model the point set is corrupted in the sense that there are N_0 outliers assumed to be distributed independently and uniformly at random on the unit sphere.

5.2 What makes clustering hard?

Two important parameters fundamentally affect the performance of subspace clustering algorithms: (1) the distance between subspaces and (2) the number and distribution of samples on each subspace.

5.2.1 Distance/affinity between subspaces

Intuitively, any subspace clustering algorithm operating on corrupted data will have difficulty segmenting observations when the subspaces are close to each other. Please see Figure 5.1 for a pictorial explanation.

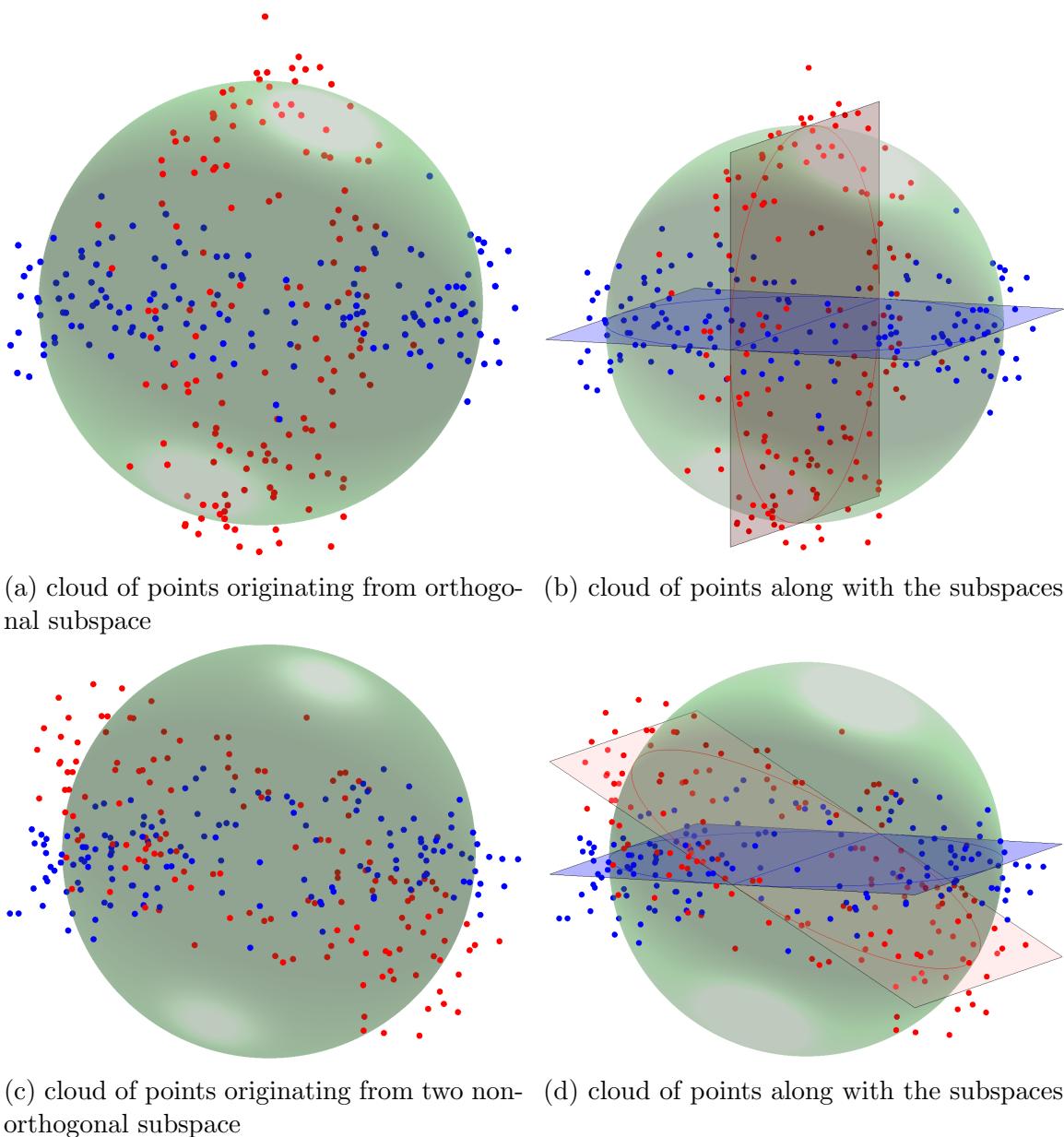
We of course need to quantify closeness, and Definition 5.2.2 captures a notion of distance or similarity/affinity between subspaces.

Definition 5.2.1 *The principal angles $\theta^{(1)}, \dots, \theta^{(d \wedge d')}$ between two subspaces S and S' of dimensions d and d' , are recursively defined by*

$$\cos(\theta^{(i)}) = \max_{\mathbf{u}_i \in S} \max_{\mathbf{v}_i \in S'} \frac{\mathbf{u}_i^T \mathbf{v}_i}{\|\mathbf{u}_i\|_{\ell_2} \|\mathbf{v}_i\|_{\ell_2}}$$

with the orthogonality constraints $\mathbf{u}_i^T \mathbf{u}_j = 0$, $\mathbf{v}_i^T \mathbf{v}_j = 0$, $j = 1, \dots, i-1$.

Alternatively, if the columns of \mathbf{U} and \mathbf{V} are orthobases for S and S' , then the cosine of the principal angles are the singular values of $\mathbf{U}^T \mathbf{V}$.



(a) cloud of points originating from orthogonal subspace (b) cloud of points along with the subspaces

(c) cloud of points originating from two non-orthogonal subspaces (d) cloud of points along with the subspaces

Figure 5.1: This picture shows how the distance between the subspaces affects the subspace clustering problem. Parts (a) and (c) depict a cloud of points. Parts (b) and (d) show the same points along with the subspaces they originate from. One can see visually that inferring the subspaces from the points is easier for parts (a) and (b) when compared with parts (c) and (d). This suggests that subspace clustering is more difficult when the subspaces are more aligned with each other.

Definition 5.2.2 *The normalized affinity between two subspaces is defined by*

$$\text{aff}(S, S') = \sqrt{\frac{\cos^2 \theta^{(1)} + \dots + \cos^2 \theta^{(d \wedge d')}}{d \wedge d'}}.$$

The affinity is a measure of correlation between subspaces. It is low when the principal angles are nearly right angles (it vanishes when the two subspaces are orthogonal) and high when the principal angles are small (it takes on its maximum value equal to one when one subspace is contained in the other). Hence, when the affinity is high, clustering is hard whereas it becomes easier as the affinity decreases. Ideally, we would like our algorithm to be able to handle higher affinity values—as close as possible to the maximum possible value.

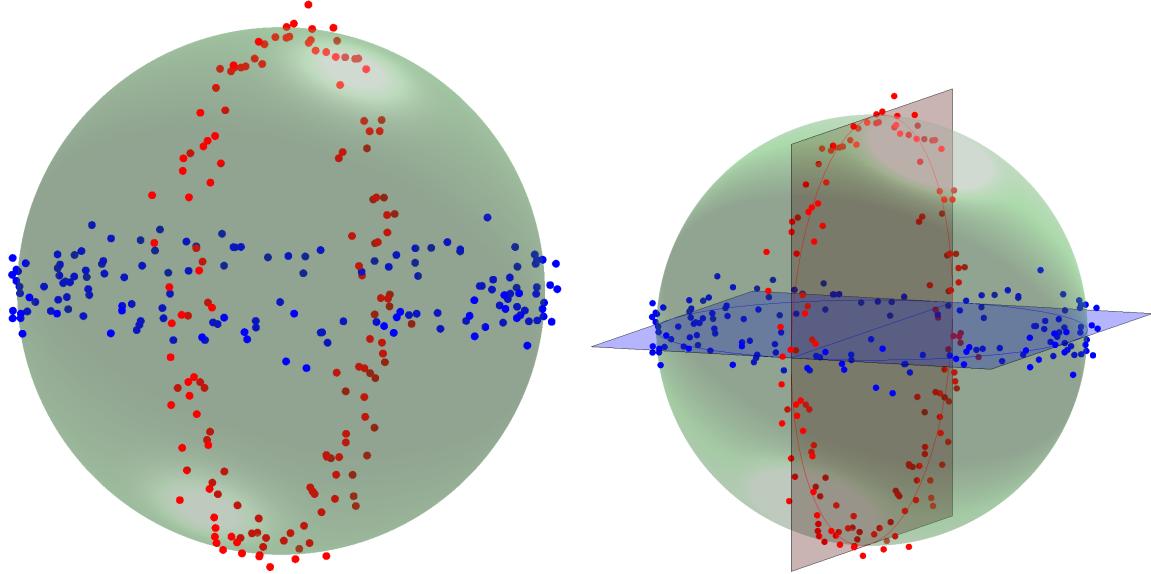
There is a statistical description of the affinity which goes as follows: sample independently two unit-normed vectors \mathbf{x} and \mathbf{y} uniformly at random from S and S' . Then

$$\mathbb{E}\{(\mathbf{x}^T \mathbf{y})^2\} \propto \{\text{aff}(S, S')\}^2,$$

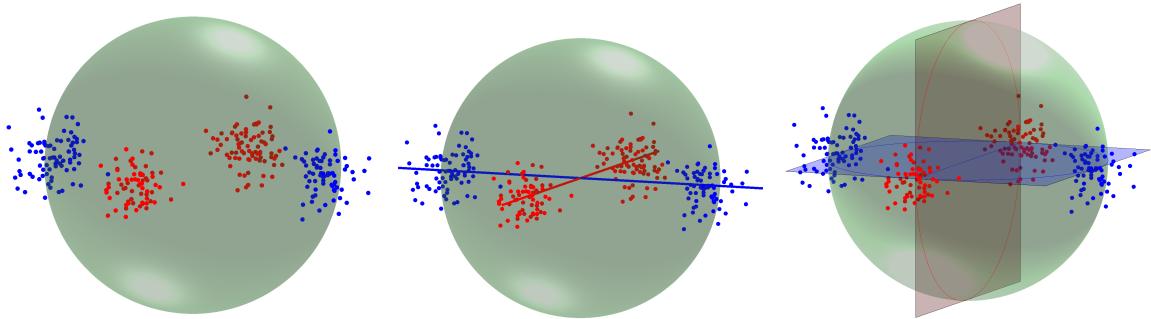
where the constant of proportionality is $d \vee d'$. Having said this, there are of course other ways of measuring the affinity between subspaces; for instance, by taking the cosine of the first principal angle. We prefer the definition above as it offers the flexibility of allowing for some principal angles to be small or zero. As an example, suppose we have a pair of subspaces with a nontrivial intersection. Then $|\cos \theta^{(1)}| = 1$ regardless of the dimension of the intersection whereas the value of the affinity would depend upon this dimension.

5.2.2 Distribution of points on each subspace and sampling density

Another important factor affecting the performance of subspace clustering algorithms has to do with the distribution of points on each subspace. In a general deterministic model where the points have arbitrary orientations on each subspace, we can imagine that the clustering problem becomes harder as the points align along even lower dimensional structures. Please see Figure 5.2 for a pictorial explanation.



(a) cloud of points that are well distributed (b) cloud of points along with the subspaces
on two subspaces



(c) cloud of points that are (d) cloud of points along with (e) cloud of points along with
badly distributed on two sub- two fitted lines two fitted planes
spaces

Figure 5.2: This picture shows how the distribution of the points on each of the subspaces affects the subspace clustering problem. Parts (a) and (c) depict a cloud of points. Parts (b), (d) and (e) show possible subspace fits. One can see visually that inferring the subspaces from the points is easier for parts (a) and (b). Inferring the subspaces in parts (c)-(e) is significantly more challenges as it is not clear which of the two alternatives (part (d) vs. part(e)) is the correct choice. This suggests that subspace clustering is more difficult when the points on the subspaces align along lower dimensional subspaces.

Later on in this chapter we shall provide appropriate measures that capture the distribution of points on each subspace. However, as most of this chapter is about the semi-random model we shall now focus on this model. In the semi-random model, the distribution of the points is essentially characterized by the number of points that lie on each subspace.

Definition 5.2.3 *The sampling density ρ of a subspace is defined as the number of samples on that subspace per dimension. In our multi-subspace model the density of S_ℓ is, therefore, $\rho_\ell = N_\ell/d_\ell$.*¹

One expects the clustering problem to become easier as the sampling density increases. Obviously, if the sampling density of a subspace S is smaller than one, then any algorithm will fail in identifying that subspace correctly as there are not sufficiently many points to identify all the directions spanned by S . Hence, we would like a clustering algorithm to be able to operate at values of the sampling density as low as possible, i.e. as close to one as possible.

5.3 Noiseless data

In this section, we shall give sufficient conditions in the fully deterministic, semi-random and fully-random models under which the SSC algorithm succeeds. Before we explain our results, we introduce some basic notation. As stated previously, we will arrange the N_ℓ points on subspace S_ℓ as columns of the matrix $\mathbf{X}^{(\ell)}$. For $\ell = 1, \dots, L$, $i = 1, \dots, N_\ell$, we use $\mathbf{X}_{(-i)}^{(\ell)}$ to denote all points on subspace S_ℓ excluding the i th point, $\mathbf{X}_{(-i)}^{(\ell)} = [\mathbf{x}_1^{(\ell)}, \dots, \mathbf{x}_{i-1}^{(\ell)}, \mathbf{x}_{i+1}^{(\ell)}, \dots, \mathbf{x}_{N_\ell}^{(\ell)}]$. We use $\mathbf{U}^{(\ell)} \in \mathbb{R}^{n \times d_\ell}$ to denote an arbitrary orthonormal basis for S_ℓ . This induces a factorization $\mathbf{X}^{(\ell)} = \mathbf{U}^{(\ell)} \mathbf{A}^{(\ell)}$, where $\mathbf{A}^{(\ell)} = [\mathbf{a}_1^{(\ell)}, \dots, \mathbf{a}_{N_\ell}^{(\ell)}] \in \mathbb{R}^{d_\ell \times N_\ell}$ is a matrix of coordinates with unit-norm columns. For any matrix $\mathbf{X} \in \mathbb{R}^{n \times N}$, the shorthand notation $\mathcal{P}(\mathbf{X})$ denotes the symmetrized convex hull of its columns, $\mathcal{P}(\mathbf{X}) = \text{conv}(\pm \mathbf{x}_1, \pm \mathbf{x}_2, \dots, \pm \mathbf{x}_N)$. Also \mathcal{P}_{-i}^ℓ stands for $\mathcal{P}(\mathbf{X}_{(-i)}^{(\ell)})$. Finally, $\|\mathbf{X}\|$ is the operator norm of \mathbf{X} and $\|\mathbf{X}\|_{\ell_\infty}$ the maximum absolute value of its entries.

¹Throughout, we take $\rho_\ell \leq e^{d_\ell/2}$. Our results hold for all other values by substituting ρ_ℓ with $\rho_\ell \wedge e^{d_\ell/2}$ in all the expressions.

5.3.1 Deterministic model

We first introduce some basic concepts needed to state our deterministic result.

Definition 5.3.1 (dual point) Consider a vector $\mathbf{y} \in \mathbb{R}^d$ and a matrix $\mathbf{A} \in \mathbb{R}^{d \times N}$, and let \mathcal{C}^* be the set of optimal solutions to

$$\max_{\vartheta \in \mathbb{R}^d} \langle \mathbf{y}, \vartheta \rangle \quad \text{subject to} \quad \|\mathbf{A}^T \vartheta\|_{\ell_\infty} \leq 1.$$

The dual point $\vartheta(\mathbf{y}, \mathbf{A}) \in \mathbb{R}^d$ is defined as a point in \mathcal{C}^* with minimum Euclidean norm.² A geometric representation is shown in Figure 5.3.

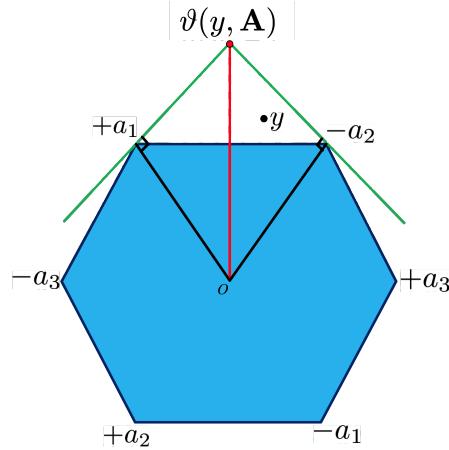


Figure 5.3: Geometric representation of a dual point, see Definition 5.3.1.

We will clarify the geometric interpretation of the dual point in Appendix A.

Definition 5.3.2 (dual directions) Define the dual directions $\mathbf{v}_i^{(\ell)} \in \mathbb{R}^n$ corresponding to the dual points $\vartheta_i^{(\ell)} = \vartheta(\mathbf{a}_i^{(\ell)}, \mathbf{A}_{(-i)}^{(\ell)})$ (arranged as columns of a matrix $\mathbf{V}^{(\ell)}$) as

$$\mathbf{v}_i^{(\ell)} = \mathbf{U}^{(\ell)} \frac{\vartheta_i^{(\ell)}}{\|\vartheta_i^{(\ell)}\|_{\ell_2}}.$$

²If this point is not unique, take $\vartheta(\mathbf{y}, \mathbf{A})$ to be any one of them.

The dual direction $\mathbf{v}_i^{(\ell)}$, corresponding to the point $\mathbf{x}_i^{(\ell)}$, from subspace S_ℓ is shown in Figure 5.4.

Definition 5.3.3 (inradius) The inradius of a convex body \mathcal{P} , denoted by $r(\mathcal{P})$, is defined as the radius of the largest Euclidean ball inscribed in \mathcal{P} .

Definition 5.3.4 (subspace incoherence) The subspace incoherence of a point set \mathcal{X}_ℓ vis a vis the other points is defined by

$$\mu(\mathcal{X}_\ell) = \max_{\mathbf{x} \in \mathcal{X} \setminus \mathcal{X}_\ell} \left\| \mathbf{V}^{(\ell)\top} \mathbf{x} \right\|_{\ell_\infty},$$

where $\mathbf{V}^{(\ell)}$ is as in Definition 5.3.2.

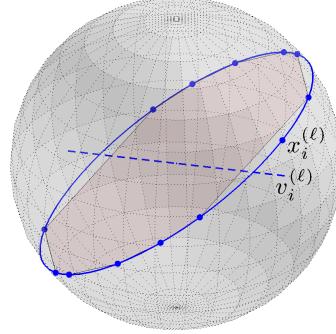


Figure 5.4: Geometric representation of a dual direction. The dual direction is the dual point embedded in the ambient n -dimensional space.

Theorem 5.3.5 If

$$\mu(\mathcal{X}_\ell) < \min_{i: \mathbf{x}_i \in \mathcal{X}_\ell} r(\mathcal{P}_{-i}^\ell) \quad (5.3.1)$$

for each $\ell = 1, \dots, L$, then the subspace detection property holds. If (5.3.1) holds for a given ℓ , then a local subspace detection property holds in the sense that for all \mathbf{x}_i , the solution to (4.1.1) has nonzero entries only when the corresponding columns of \mathbf{X} are in the same subspace as \mathbf{x}_i .

The incoherence parameter between a set of points on one subspace and the set of dual directions on another, is a measure of affinity between the two subspaces.

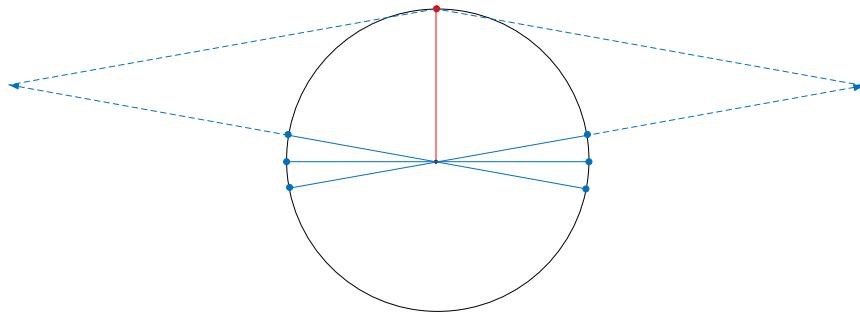


Figure 5.5: Skewed distribution of points on a single subspace and ℓ_1 synthesis.

To see why, notice that if the incoherence between two subspaces is high, it implies that there is a point on one subspace and a direction on another (a dual direction) such that the angle between them is small. That is, the subspaces are ‘close’, hence, clustering becomes hard. The inradius measures the spread of points. A very small minimum inradius implies that the distribution of points is skewed towards certain directions; thus, *subspace* clustering using an ℓ_1 penalty is difficult. To see why this is so, assume the subspace is of dimension 2 and all of the points on the subspace are skewed towards one line, except for one “special” point which is in the direction orthogonal to that line. This is shown in Figure 5.5 with the “special” point in red and the others in blue. To synthesize this special point as a linear combination of the other points from its subspace, we would need huge coefficient values and this is why it may very well be more economical—in an ℓ_1 sense—to select points from other subspaces. This is a situation where ℓ_0 minimization would still be successful but its convex surrogate is not (researchers familiar with sparse regression would recognize a setting in which variables are correlated, and which is challenging for the LASSO.) Theorem 5.3.5 essentially states that as long as different subspaces are not similarly oriented and the points on a single subspace are well spread, SSC can cluster the data correctly. A geometric perspective of (5.3.1) is provided in Appendix A.

To get concrete results, one needs to estimate both the incoherence and inradius in terms of the parameters of interest, which include the number of subspaces, the dimensions of the subspaces, the number of points on each subspace, and so on. To do this, we use the probabilistic models we introduced earlier. This is our next topic.

5.3.2 Semi-random model

In case the distribution of the points are uniform on their corresponding subspaces, the Geometric Condition (5.3.1) reduces to a simple condition, which holds with high probability. This is the subject of the next theorem.

Our results make use of a constant $c(\rho)$ obeying the following two properties:

- (i) For all $\rho > 1$, $c(\rho) > 0$.
- (ii) There is a numerical value ρ^* , such that for all $\rho \geq \rho^*$, one can take $c(\rho) = \frac{1}{\sqrt{8}}$.

Theorem 5.3.6 *Suppose $N_\ell = \rho_\ell d_\ell + 1$ points are chosen on each subspace S_ℓ at random, $1 \leq \ell \leq L$. Then as long as*

$$\max_{k:k \neq \ell} 4\sqrt{2} \left(\log[N_\ell(N_k + 1)] + \log L + t \right) \text{aff}(S_k, S_\ell) < c(\rho_\ell) \sqrt{\log \rho_\ell}, \quad \text{for each } \ell \quad (5.3.2)$$

the subspace detection property holds with probability at least

$$1 - \sum_{\ell=1}^L N_\ell e^{-\sqrt{d_\ell} \sqrt{N_\ell - 1}} - \frac{1}{L^2} \sum_{k \neq \ell} \frac{4e^{-2t}}{(N_k + 1) N_\ell}.$$

Hence, ignoring log factors, subspace clustering is possible if the affinity between the subspaces is less than a sufficiently small constant.

To derive useful results, assume for simplicity that we have L subspaces of the same dimension d and $\rho d + 1$ points per subspace so that $N = L(\rho d + 1)$. Then perfect clustering occurs with probability at least $1 - Ne^{-\sqrt{\rho d}} - \frac{2}{(\rho d)(\rho d + 1)} e^{-2t}$ if

$$\text{aff}(S_k, S_\ell) < \frac{c(\rho) \sqrt{\log \rho}}{4\sqrt{2}(2 \log N + t)}. \quad (5.3.3)$$

Our notion of affinity matches our basic intuition. To be sure, if the subspaces are too close to each other (in terms of our defined notion of affinity), subspace clustering is hard. Having said this, our result has an element of surprise. Indeed, the affinity can at most be one and, therefore, our result essentially states that if the affinity is

less than a sufficiently small constant, then SSC works. Now this allows for subspaces to intersect and, yet, SSC still provably clusters all the data points correctly!

To discuss other aspects of this result, assume as before that all subspaces have the same dimension d . When d is small and the total number of subspaces is $\mathcal{O}(n/d)$, the problem is inherently hard because it involves clustering all the points into many small subgroups. This is reflected by the low probability of success in Theorem 5.3.6. Of course if one increases the number of points chosen from each subspace, the problem should intuitively become easier. The probability associated with (5.3.3) allows for such a trend. In other words, when d is small, one can increase the probability of success by increasing ρ . Introducing a parameter $0 < \beta \leq 1$, the condition can be modified to

$$\text{aff}(S_k, S_\ell) < \frac{c(\rho)\sqrt{\beta \log \rho}}{4(2 \log N + t)}, \quad (5.3.4)$$

which holds with probability at least $1 - Ne^{-\rho^{(1-\beta)}d} - \frac{2}{(\rho d)(\rho d+1)}e^{-2t}$. The more general condition (5.3.2) and the corresponding probability can also be modified in a similar manner.

5.3.3 Fully-random model

We now turn our attention to the fully random model. We establish that the subspace detection property holds as long as the dimensions of the subspaces are roughly linear in the ambient dimension. Put differently, SSC can provably achieve perfect subspace recovery in settings not previously understood.

Theorem 5.3.7 *Assume there are L subspaces, each of dimension d , chosen independently and uniformly at random. Furthermore, suppose there are $\rho d + 1$ points chosen independently and uniformly at random on each subspace.³ Then the subspace*

³From here on, when we say that points are chosen from a subspace, we implicitly assume they are unit normed. For ease of presentation we state our results for $1 < \rho \leq e^{\frac{d}{2}}$, i.e. the number of points on each subspace is not exponentially large in terms of the dimension of that subspace. The results hold for all $\rho > 1$ by replacing ρ with $\min\{\rho, e^{\frac{d}{2}}\}$.

detection property holds with large probability as long as

$$d < \frac{c^2(\rho) \log \rho}{12 \log N} n \quad (5.3.5)$$

($N = L(\rho d + 1)$ is the total number of data points). The probability is at least $1 - \frac{2}{N} - Ne^{-\sqrt{\rho}d}$, which is calculated for values of d close to the upper bound. For lower values of d , the probability of success is of course much higher, as explained below.

This theorem conforms to our intuition since clustering becomes more difficult as the dimensions of the subspaces increase. Intuitively, another difficult regime concerns a situation in which we have very many subspaces of small dimensions. This difficulty is reflected in the dependence of the denominator in (5.3.5) on L , the number of subspaces (through N).

As it becomes clear in the proof (see Section 7.2), a slightly more general version of Theorem 5.3.7 holds, namely, with $0 < \beta \leq 1$, the subspace detection property holds as long as

$$d < 2\beta \left[\frac{c^2(\rho) \log \rho}{12 \log N} \right] n \quad (5.3.6)$$

with probability at least $1 - \frac{2}{N} - Ne^{-\rho^{(1-\beta)}d}$. Therefore, if d is a small fraction of the right-hand side in (5.3.5), the subspace detection property holds with much higher probability, as expected.

An interesting regime is when the number of subspaces L is fixed and the density of points per subspace is $\rho = d^\eta$, for a small $\eta > 0$. Then as $n \rightarrow \infty$ with the ratio d/n fixed, it follows from $N \asymp L\rho d$ and (5.3.6) using $\beta = 1$ that the subspace detection property holds as long as

$$d < \frac{\eta}{48(1+\eta)} n.$$

This justifies our earlier claims since we can have subspace dimensions growing linearly in the ambient dimension. It should be noted that this asymptotic statement is only a factor 8–10 away from what is observed in simulations, which demonstrates a relatively small gap between our theoretical predictions and simulations.⁴

⁴To be concrete, when the ambient dimension is $n = 50$ and the number of subspaces is $L = 10$, the subspace detection property holds for d in the range from 7 to 10.

5.3.4 Comparison with previous results on SSC

In this section we will review existing conditions involving a restriction on the minimum angle between subspaces under which SSC is expected to work. Our contribution with respect to these results is that we show SSC works in much broader situations.

In [95], Elhamifar and Vidal show that under less restrictive conditions the ℓ_1 subspace detection property still holds. Formally, they show that if

$$\frac{1}{\sqrt{d_\ell}} \max_{\mathbf{Y} \in \mathbb{W}_{d_\ell}(\mathbf{X}^{(\ell)})} \sigma_{\min}(\mathbf{Y}) > \max_{k: k \neq \ell} \cos(\theta_{k\ell}^{(1)}) \quad \text{for all } \ell = 1, \dots, L, \quad (5.3.7)$$

then the subspace detection property holds. In the above formulation, $\sigma_{\min}(\mathbf{Y})$ denotes the smallest eigenvalue of \mathbf{Y} and $\mathbb{W}_d(\mathbf{X}^{(\ell)})$ denotes the set of all full rank sub-matrices of $\mathbf{X}^{(\ell)}$ of size $n \times d_\ell$. The interesting part of the above condition is the appearance of the principal angle on the right-hand side. However, the left-hand side is not particularly insightful (i.e. it does not tell us anything about the important parameters involved in the subspace clustering problem, such as dimensions, number of subspaces, and so on.) and it is in fact NP-hard to even calculate it.

- **Deterministic model.** This paper also introduces a sufficient condition (5.3.1) under which the subspace detection property holds in the fully deterministic setting, compare Theorem 5.3.5. This sufficient condition is much less restrictive as any configuration obeying (5.3.7) also obeys (5.3.1). As for (5.3.7), checking that (5.3.1) holds is also NP-hard in general. However, to prove that the subspace detection property holds, it is sufficient to check a slightly less restrictive condition than (5.3.1); this is tractable, see Lemma 7.2.1.
- **Semi-random model.** Assume that all subspaces are of the same dimension d and that there are $\rho d + 1$ points on each subspace. Since the columns of \mathbf{Y} have unit norm, it is easy to see that the left-hand side of (5.3.7) is strictly less than $1/\sqrt{d}$. Thus, (5.3.7) at best restricts the range for perfect subspace recovery to $\cos \theta < c \frac{1}{\sqrt{d}}$ (by looking at (5.3.7), it is not entirely clear that this would even

be achievable). In comparison, Theorem 5.3.6 requires

$$\text{aff}(S_k, S_\ell) = \frac{\sqrt{\cos^2(\theta^{(1)}) + \cos^2(\theta^{(2)}) + \dots + \cos^2(\theta^{(d)})}}{\sqrt{d}} < c\sqrt{\log(\rho)}. \quad (5.3.8)$$

The left-hand side of can be much smaller than $\sqrt{d} \cos \theta$ and is, therefore, less restrictive.

To be more specific, assume that in the model described above we have two subspaces with an intersection of dimension s . Because the two subspaces intersect, the condition given by Elhamifar and Vidal becomes $1 < \frac{1}{\sqrt{d}}$, which cannot hold. In comparison, our condition (5.3.8) simplifies to

$$\cos^2(\theta^{(s+1)}) + \dots + \cos^2(\theta^{(d)}) < c \log(\kappa) d - s,$$

which holds as long as s is not too large and/or a fraction of the angles are not too small. From an application standpoint, this is important because it explains why SSC can often succeed even when the subspaces are not disjoint.

- **Fully random model.** As before, assume for simplicity that all subspaces are of the same dimension d and that there are $\rho d + 1$ points on each subspace. We have seen that (5.3.7) imposes $\cos \theta < c \frac{1}{\sqrt{d}}$. It can be shown that in the fully random setting,⁵ $\cos \theta \approx c \sqrt{\frac{d}{n}}$. Therefore, (5.3.7) would put a restriction of the form

$$d < c\sqrt{n}.$$

In comparison, Theorem 5.3.7 requires

$$d < c_1 \frac{\log \rho}{\log N} n,$$

which allows for the dimension of the subspaces to be almost linear in the ambient dimension.

⁵One can see this by noticing that the square of this parameter is the largest root of a multivariate beta distribution. The asymptotic value of this root can be calculated e.g. see [136].

5.4 Segmentation in the presence of noise

This section presents our main theoretical results concerning the performance of RSC-N (Algorithm 5). Throughout this section we assume that the clean data points are drawn based on the semi-random model. Furthermore, we make two additional assumptions:

- **Affinity condition.** We say that a subspace S_ℓ obeys the *affinity condition* if

$$\max_{k: k \neq \ell} \text{aff}(S_\ell, S_k) \leq \eta_0 / \log N, \quad (5.4.1)$$

where η_0 a fixed numerical constant.

- **Sampling condition.** We say that subspace S_ℓ obeys the *sampling condition* if

$$\rho_\ell \geq \rho^*, \quad (5.4.2)$$

where ρ^* is a fixed numerical constant.

- **Bounded noise level.** We assume

$$\sigma < \sigma^*, \quad \text{and} \quad \max_\ell d_\ell < c_0 \frac{n}{(\log N)^2}, \quad (5.4.3)$$

where $\sigma^* < 1$ and c_0 are fixed numerical constants. To remove any ambiguity, σ is the noise level and σ^* the maximum value it can take on. The second assumption is here to avoid unnecessarily complicated expressions later on. While more substantial, the first is not too restrictive since it just says that the signal \mathbf{x} and the noise \mathbf{z} may have about the same magnitude. (With an arbitrary perturbation of Euclidean norm equal to two, one can move from any point \mathbf{x} on the unit sphere to just about any other point.)

The careful reader might argue that we should require smaller affinity values as the noise level increases. The reason why σ does not appear in (5.4.1) is that we assumed a bounded noise level. For higher values of σ , the affinity condition would read as in

(5.4.1) with a right-hand side equal to

$$\eta = \frac{\eta_0}{\log N} - \sigma \sqrt{\frac{d_\ell}{2n \log N}}.$$

5.4.1 Main results

From here on we use $d(i)$ to refer to the dimension of the subspace the vector \mathbf{y}_i originates from. $N(i)$ and $\rho(i)$ are used in a similar fashion for the number and density of points on this subspace.

Theorem 5.4.1 (No false discoveries) *Assume that the subspace attached to the i th column obeys the affinity and sampling conditions and that the noise level σ is bounded as in (5.4.3), where σ^* is a sufficiently small numerical constant. In Algorithm 5, take $\tau = 2\sigma$ and $f(t)$ obeying $f(t) \geq 0.707\sigma t^{-1}$. Then with high probability,⁶ there is no false discovery in the i th column of \mathbf{B} .*

Theorem 5.4.2 (Many true discoveries) *Consider the same setup as in Theorem 5.6.3 with $f(\cdot)$ also obeying $f(t) \leq \alpha_0 t^{-1}$ for some numerical constant α_0 . Then with high probability,⁷ there are at least*

$$c_1 \frac{d(i)}{\log \rho(i)} \tag{5.4.4}$$

true discoveries in the i th column (c_1 is a positive numerical constant).

The above results indicate that the first step of RSC-N works correctly in fairly broad conditions. To give an example, assume two subspaces of dimension d overlap in a smaller subspace of dimension s but are orthogonal to each other in the remaining directions (equivalently, the first s principal angles are 0 and the rest are $\pi/2$). In this case, the affinity between the two subspaces is equal to $\sqrt{s/d}$ and (5.4.1) allows s to grow almost linearly in the dimension of the subspaces. Hence, subspaces can have intersections of large dimensions.

⁶probability at least $1 - 2e^{-\gamma_1 n} - 6e^{-\gamma_2 d(i)} - e^{-\sqrt{N(i)d(i)}} - \frac{23}{N^2}$, for fixed numerical constants γ_1, γ_2 .

⁷probability at least $1 - 2e^{-\gamma_1 n} - 6e^{-\gamma_2 d(i)} - e^{-\sqrt{N(i)d(i)}} - \frac{23}{N^2}$, for fixed numerical constants γ_1, γ_2 .

In the noiseless case, we have already shown in Theorem 5.3.6 that when the sampling condition holds and

$$\max_{k:k \neq \ell} \text{aff}(S_\ell, S_k) \leq \eta_0 \frac{\sqrt{\log \rho_\ell}}{\log N},$$

(albeit with slightly different values η_0 and ρ^*), then applying the noiseless version (4.1.1) of the algorithm also yields no false discoveries. Hence, with the proviso that the noise level is not too large, conditions under which the algorithm is provably correct are essentially the same.

Earlier, we argued that we would like to have, if possible, an algorithm provably working at (1) high values of the affinity parameters and (2) low values of the sampling density as these are the conditions under which the clustering problem is challenging. (Another property on the wish list is the ability to operate properly with high noise or low SNR and this is discussed next.) In this context, since the affinity is at most one, our results state that the affinity can be within a log factor from this maximum possible value. The number of samples needed per subspace is minimal as well. That is, as long as the density of points on each subspace is larger than a constant $\rho > \rho^*$, the algorithm succeeds.⁸

We would like to have a procedure capable of making no false discoveries and many true discoveries at the same time. Now in the noiseless case, whenever there are no false discoveries, the i th column contains exactly $d(i)$ true discoveries. Theorem 5.6.4 states that as long as the noise level σ is less than a fixed numerical constant, the number of true discoveries is roughly on the same order as in the noiseless case. In other words, a noise level of this magnitude does not fundamentally affect the performance of the algorithm. This holds even when there is great variation in the dimensions of the subspaces, and is possible because λ is appropriately tuned in an adaptive fashion.

The number of true discoveries is shown to scale at least like dimension over the log of the density. This may suggest that the number of true discoveries decreases

⁸This is with the proviso that the density does not grow exponentially in the dimension of the subspace. This is not a restrictive assumption as having exponentially many points from the same subspace makes the problem especially easy.

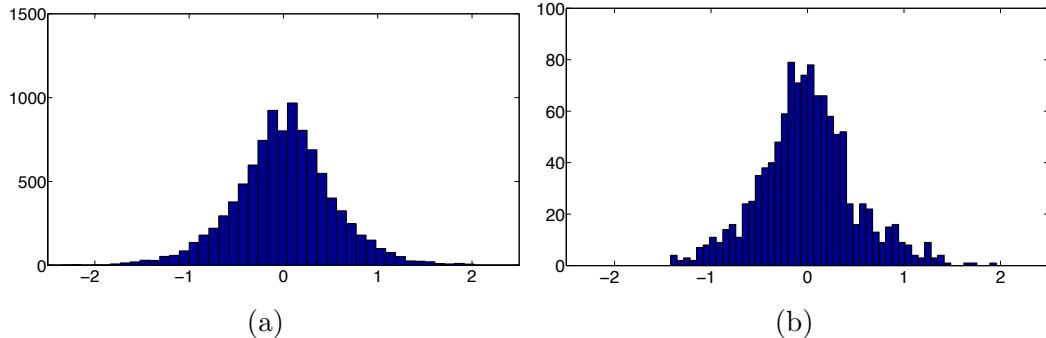


Figure 5.6: Histograms of the true discovery values from the two step procedure with $\alpha_0 = 0.25$ (multiplied by \sqrt{d}). (a) $d = 200$. (b) $d = 20$.

(albeit very slowly) as the sampling density increases. This behavior is to be expected: when the sampling density becomes exponentially large (in terms of the dimension of the subspace) the number of true discoveries become small since we need fewer columns to synthesize a point. In fact, the $d/\log \rho$ behavior seems to be the correct scaling. Indeed, when the density is low and ρ takes on a small value, (5.6.4) asserts that we make on the order of d discoveries, which is tight. Imagine now that we are in the high-density regime and ρ is exponential in d . Then as the points gets tightly packed, we expect to have only one discovery in accordance with (5.6.4).

Theorem 5.6.4 establishes that there are many true discoveries. This would not be useful for clustering purposes if there were only a handful of very large true discoveries and all the others of negligible magnitude. The reason is that the similarity matrix \mathbf{W} would then be close to a sparse matrix and we would run the risk of splitting true clusters. Our proofs show that this does not happen although we do not present an argument for lack of space. Rather, we demonstrate this property empirically. On our running example, Figures 5.6a and 5.6b show that the histograms of appropriately normalized true discovery values resemble a bell-shaped curve. Note that each true discovery corresponds to a non-zero coefficient which can take on either a positive or negative value.

Finally, we would like to comment on the fact that our main results hold when λ belongs to a fairly broad range of values. First, when all the subspaces have small dimensions, one can choose the same value of λ for all the data points since $1/\sqrt{d}$ is

essentially constant. Hence, when we know a priori that we are in such a situation, there may be no need for the two step procedure. (We would still recommend the conservative two-step procedure because of its superior empirical performance on real data.) Second, the proofs also reveal that if we have knowledge of the dimension of the largest subspace d_{\max} , the first theorem holds with a fixed value of λ proportional to $\sigma/\sqrt{d_{\max}}$. Third, when the subspaces themselves are drawn at random, the first theorem holds with a fixed value of λ proportional to $\sigma(\log N)/\sqrt{n}$. (Both these statements follow by plugging these values of λ in the proofs of Section 7.3 and we omit the calculations.) We merely mention these variants to give a sense of what our theorems can also give. As explained earlier, we recommend the more conservative two-step procedure with the proxy for $1/\sqrt{d}$. The reason is that using a higher value of λ allows for a larger value of η_0 in (5.4.1), which says that the subspaces can be even closer. In other words, we can function in a more challenging regime. To drive this point home, consider the noiseless problem. When the subspaces are close, the equality constrained ℓ_1 problem may yield some false discoveries. However, if we use the LASSO version—even though the data is noiseless—we may end up with no false discoveries while maintaining sufficiently many true discoveries.

5.5 Segmentation with gross outliers

To see how RSC-O (Algorithm 6) works in the presence of outliers, we begin by introducing a proper threshold function, and define

$$\kappa(\gamma) = \begin{cases} \sqrt{\frac{2}{\pi}} \frac{1}{\sqrt{\gamma}}, & \text{if } 1 \leq \gamma \leq e, \\ \sqrt{\frac{2}{\pi e}} \frac{1}{\sqrt{\log \gamma}}, & \text{if } \gamma \geq e, \end{cases} \quad (5.5.1)$$

shown in Figure 5.7. We shall present two theorems which collectively justify that RSC-O is stable vis à vis gross outliers. Our first result asserts that as long as the number of outliers is not overwhelming, RSC-O detects all of them.

Theorem 5.5.1 *Assume there are N_d points to be clustered together with N_0 outliers sampled uniformly at random on the $n - 1$ -dimensional unit sphere ($N = N_0 + N_d$).*

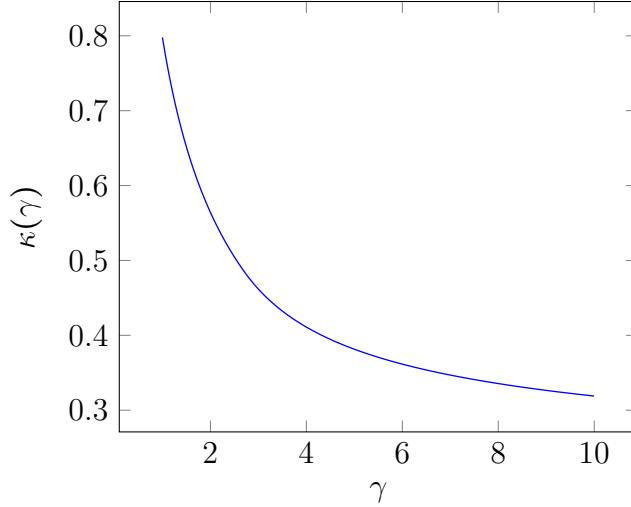


Figure 5.7: Plot of the threshold function (5.5.1).

RSC-O detects all of the outliers with high probability⁹ as long as

$$N_0 < \frac{1}{n} e^{c\sqrt{n}} - N_d,$$

where c is a numerical constant. Furthermore, suppose the subspaces are d -dimensional and of arbitrary orientation, and that each contains $\rho d + 1$ points sampled independently and uniformly at random. Then with high probability,¹⁰ Algorithm 6 does not detect any subspace point as outlier provided that

$$N_0 < n\rho^{c_2 \frac{n}{d}} - N_d$$

in which $c_2 = c^2(\rho)/(2e^2\pi)$.

This result shows that our outlier detection scheme can reliably detect all outliers even when their number grows exponentially in the root of the ambient dimension. We emphasize that this holds without making any assumption whatsoever about the orientation of the subspaces or the distribution of the points on each subspace.

⁹With probability at least $1 - N_0 e^{-Cn/\log(N_0+N_d)}$. If $N_0 < \frac{1}{n} e^{c\sqrt{n}} - N_d$, this is at least $1 - \frac{1}{n}$.

¹⁰With probability at least $1 - N_0 e^{-Cn/\log(N_0+N_d)} - N_d e^{-\sqrt{\rho}d}$. If $N_0 < \min\{n e^{c_2 \frac{n}{d}}, \frac{1}{n} e^{c\sqrt{n}}\} - N_d$, this is at least $1 - \frac{1}{n} - N_d e^{-\sqrt{\rho}d}$.

Furthermore, if the points on each subspace are uniformly distributed, our scheme will not wrongfully detect a subspace point as an outlier. In the next theorem we show that similar results hold under less restrictive assumptions.

Theorem 5.5.2 *Suppose the outlier points are chosen uniformly at random and set $\gamma = \frac{N-1}{n}$, then using the threshold value $(1-t)\frac{\kappa(\gamma)}{\sqrt{e}}\sqrt{n}$, all outliers are identified correctly with probability at least $1 - N_0 e^{-C_1 t^2 \frac{n}{\log N}}$ for some positive numerical constant C_1 . Furthermore, we have the following guarantees in the deterministic and semi-random models.*

(a) *If in the deterministic model,*

$$\max_{\ell,i} \frac{1}{r(\mathcal{P}(\mathbf{X}_{(-i)}^{(\ell)}))} < (1-t)\frac{\kappa(\gamma)}{\sqrt{e}}\sqrt{n}, \quad (5.5.2)$$

then no ‘real’ data point is wrongfully detected as an outlier.

(b) *If in the semi-random model,*

$$\max_{\ell} \frac{\sqrt{2d_{\ell}}}{c(\rho_{\ell})\sqrt{\log \rho_{\ell}}} < (1-t)\frac{\kappa(\gamma)}{\sqrt{e}}\sqrt{n}, \quad (5.5.3)$$

then with probability at least $1 - \sum_{\ell=1}^L N_{\ell} e^{-\sqrt{d_{\ell}}\sqrt{(N_{\ell}-1)}}$, no ‘real’ data point is wrongfully detected as an outlier.

The threshold in the right-hand side of (5.5.2) and (5.5.3) is essentially \sqrt{n} multiplied by a factor which depends only on the ratio of the number of points and the dimension of the ambient space.

As in the situation with no outliers, when d_{ℓ} is small we need to increase N_{ℓ} to get a result holding with high probability. Again this is expected because when d_{ℓ} is small, we need to be able to separate the outliers from many small clusters which is inherently a “hard” problem for small values of N_{ℓ} .

The careful reader will notice a factor \sqrt{e} discrepancy between the threshold $\kappa(\gamma)\sqrt{n}$ presented in Algorithm 6 and what is proven in (5.5.2) and (5.5.3). We

believe that this is a result of our analysis¹¹ and we conjecture that (5.5.2) and (5.5.3) hold without the factor \sqrt{e} in the denominator. Our simulations in Section 6.2.2 support this conjecture.

5.5.1 Comparison with other theoretical results

We pause to compare our results regarding outlier detection with a few other theoretical results. In [?], Lerman and Zhang study the effectiveness of recovering subspaces in the presence of outliers by some sort of ℓ_p minimization for different values of $0 < p < \infty$. They address simultaneous recovery of all L subspaces by minimizing the functional

$$e_{\ell_p}(\mathcal{X}, S_1, \dots, S_L) = \sum_{x \in \mathcal{X}} \min_{1 \leq \ell \leq L} (\text{dist}(x, S_\ell))^p. \quad (5.5.4)$$

Here, S_1, \dots, S_L are the optimization variables and \mathcal{X} is our data set. This is not a convex optimization for any $p > 0$, since the feasible set is the Grassmannian.

In the semi-random model, the result of Lerman and Zhang states that under the assumptions stated in Theorem 5.5.1, with $0 < p \leq 1$ and τ_0 a constant,¹² the subspaces S_1, \dots, S_L minimize (with large probability) the energy (5.5.4) among all d -dimensional subspaces in \mathbb{R}^n if

$$N_0 < \tau_0 \rho d \min \left(1, \min_{k \neq \ell} \text{dist}(S_k, S_\ell)^p / 2^p \right). \quad (5.5.5)$$

It is easy to see that the right-hand side of (5.5.5) is upperbounded by ρd , i.e. the typical number of points on each subspace. Notice that our analogous result in Theorem 5.3.7 allows for a much larger number of outliers. In fact, the number of outliers can sometimes even be much larger than the total number of data points on all subspaces combined. Our proposed algorithm also has the added benefit that it

¹¹More specifically, from switching from the mean width to a volumetric argument by means of Urysohn's inequality.

¹²The result of [?] is a bit more general in that the points on each subspace can be sampled from a single distribution obeying certain regularity conditions, other than the uniform measure. In this case, τ_0 depends on this distribution as well.

is convex and, therefore, practical. Having said this, it is worth mentioning that the results in [?] hold for a more general outlier model. Also, an interesting byproduct of the result from Lerman and Zhang is that the energy minimization can perform perfect subspace recovery when no outliers are present. In fact, they even extend this to the case when the subspace points are noisy.

After the publication of our results [150] also addressed outlier detection. However, the suggested scheme limits the number of outliers to $N_0 < n - \sum_{\ell=1}^L d_\ell$. That is, when the total dimension of the subspaces ($\sum_{\ell=1}^L d_\ell$) exceeds the ambient dimension n , outlier detection is not possible based on the suggested scheme. In contrast, our results guarantee perfect outlier detection even when the number of outliers far exceeds the number of data points.

5.6 Towards segmentation with missing data

In this section we present some theoretical results and conjectures that indicate the RSC-M algorithm is effective for subspace clustering with missing data. Through out this section we shall assume that the clean data points are drawn according to the semi-random model. Before we present our results we first explore conditions under which subspace clustering with missing data is possible.

5.6.1 When is subspace clustering with missing data possible?

As explained in Section 5.2 any subspace clustering algorithm, even when the data points are free from any form of corruption, will have difficulty segmenting data points when the subspaces are close to each other or when there are not sufficient number of sample points available from each subspace. Thus to overcome this difficulty we assume that the affinity and sampling conditions (conditions (5.4.1) and (5.4.2)) hold for the clean data points. However, subspace clustering with missing data presents additional challenges. We shall explore these challenges next.

5.6.1.1 What fraction of missing entries?

Suppose that the number of columns from the different subspaces are represented by the L-tuple $(N_1, N_2, \dots, N_L) = (\rho_1 d_1, \rho_2 d_2, \dots, \rho_L d_L)$. Such a matrix can be represented by the numbers $(nN_1, nN_2, \dots, nN_L)$, but it only has the following degrees of freedom $((2n - d_1)d_1, (2n - d_2)d_2, \dots, (2n - d_L)d_L)$. This can be easily seen by counting parameters in the singular value decomposition of the samples from each of the subspaces. Therefore the number of samples we get to see from each subspace must be larger than the degrees of freedom associated with that subspace. This implies the following number of samples per column,

$$n(1 - \delta) \geq \frac{(2n - d_\ell)}{\rho_\ell}. \quad (5.6.1)$$

This simple calculation suggests that subspace clustering with missing data becomes easier as the ambient dimension and the density of points per subspace (ρ_ℓ) increases, and harder as the fraction of missingness (δ) increases. To ensure that (5.6.1) holds, in addition to the sampling condition (5.4.2) we also assume the following condition holds

- **Bounded fraction of missing entries.** We assume that

$$\delta < \delta^*, \quad (5.6.2)$$

where $\delta^* < 1$ is a fixed numerical constant.

5.6.1.2 Which subspaces?

Even if the assumptions stated so far hold we cannot hope to be able to cluster/recover a matrix whose columns originate from a union of subspaces only from a sampling of its entries. Assume that one of the subspaces is the span of the first 2 canonical coordinate axes e_1, e_2 . Therefore, the submatrix corresponding to that subspace is of

the form

$$\mathbf{X}^{(1)} = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1N_1} \\ x_{21} & x_{22} & \dots & x_{2N_1} \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix}.$$

This matrix may have non-zero entries in the first two rows and all the other entries are 0. Clearly this matrix cannot be recovered from a sampling of its entries unless we see almost all of it. The reason is that for most sampling sets, we would only get to see zeros and we have no way of knowing that the matrix is not zero. This suggests that the subspaces should not be very aligned with the canonical basis vectors. this leads to the following definition. Please see [62] for a similar definition in the context of matrix completion.

Definition 5.6.1 (Incoherence) *Let $\mathbf{U} \in \mathbb{R}^{n \times d}$ be an orthonormal basis for subspace S and let \mathbf{u}_i denote its i -th row. Define*

$$\mu(S) = \sqrt{\frac{n}{d}} \max_i \|\mathbf{u}_i\|_{\ell_2}.$$

To avoid pathological cases such as the one stated above, we assume that all the subspaces obey the following condition.

- **Incoherence conditons.** We say that subspace S_ℓ obeys the *incoherence condition* if

$$\mu(S_\ell) \leq \mu^*,$$

where μ^* is a fixed numerical constant.

5.6.2 What is the correct choice of λ ?

We remind the reader that we use $d(i)$ to refer to the dimension of the subspace the vector \mathbf{y}_i originates from. $N(i)$ and $\rho(i)$ are used in a similar fashion for the

number and density of points on this subspace. Our results on the analysis of the SSC algorithm (Section 5.3) suggests that the solution of the idealized problem (4.6.1) has good clustering properties. We remind the reader that $\beta^I(i)$ is used to denote this ideal solution. That is,

$$\beta^I(i) = \arg \min_{\beta \in \mathbb{R}^N} \|\beta\|_{\ell_1} \quad \text{subject to} \quad \mathbf{X}_{\Omega_i} \beta = \mathbf{x}_{\Omega_i}^{(i)} \text{ and } \beta_i = 0. \quad (5.6.3)$$

Therefore, it makes sense to try to find a formulation with optimal solution that resembles the ideal solution $\beta^I(i)$. This suggests that we should pick λ such that $\beta^I(i)$ is feasible for the Bias-corrected Dantzig selector in (5.6.3).

Theorem 5.6.2 *Assume that the subspace attached to the i th column obeys the affinity, sampling and incoherence conditions and that the fraction of missing entries δ is bounded as in (5.6.2), where δ^* is a sufficiently small numerical constant. If we use*

$$\lambda = C \frac{(\log N)^2}{\sqrt{n}} \sqrt{\frac{\delta}{1-\delta}},$$

in the optimization problem (4.6.4) with a sufficiently large constant C , then the ideal solution $\beta^I(i)$ is feasible for the bias-corrected Dantzig selector (4.6.4) with probability at least $1 - 15/N - 5e^{-\gamma_1 d(i)} - e^{-\sqrt{d(i)N(i)}}$.

This theorem specifies the correct scaling of λ up to constant and log factors. Numerical experiments in Section 6.2.3 suggest that $\lambda = \frac{\sqrt{2 \log(nN)}}{\sqrt{n}} \sqrt{\frac{\delta}{1-\delta}}$ is a good choice. Notice that this choice of λ is essentially parameter-less as the fraction of missing entries (δ) can be easily estimated from the data as detailed in Section 4.6.1.

5.6.3 Guarantees for subspace clustering with missing data

Conjecture 5.6.3 (No false discoveries) *Assume that the subspace attached to the i th column obeys the affinity, sampling and incoherence conditions and that the bound on the fraction of missing entries is satisfied. Set*

$$\lambda = C \frac{(\log N)^2}{\sqrt{n}} \sqrt{\frac{\delta}{1-\delta}},$$

for a sufficiently large constant C . Then with high probability, there is no false discovery in the i th column of \mathbf{B} .

Conjecture 5.6.4 (Many true discoveries) Consider the same setup as in conjecture 5.6.3. Then with high probability, there are at least

$$c_0 \frac{d(i)}{\log \rho(i)} \quad (5.6.4)$$

true discoveries in the i th column (c_0 is a positive numerical constant).

If correct, these conjectures would indicate that the algorithm works correctly in fairly broad conditions. It shows that the algorithm provably works at (1) high values of the affinity parameters, (2) low values of the sampling density, and (3) high fraction of missing entries. These are the conditions under which the clustering problem is challenging. In terms of these parameters the performance of the algorithm is essentially optimal as no algorithm regardless of tractability can handle an affinity value or fraction of missing entries higher than one or require a sampling density less than one. As detailed in the previous section analogous results exists for the noiseless and noisy subspace clustering problem. Proof of the conjectures above completes this line of research and shows that essentially the same results hold even when a large fraction of the entries of the data matrices are missing.

5.7 Comparison with other schemes

In this section we shall focus on comparing theoretical properties of our approach with other leading subspace clustering techniques. Three themes will help in organizing our discussion.

- *Tractability.* Is the proposed method or algorithm computationally tractable?
- *Robustness.* Is the algorithm provably robust to noise and other imperfections?

- *Efficiency.* Is the algorithm correctly operating near the limits we have identified above? In our model, how many points do we need per subspace? How large can the affinity between subspaces be?

We discussed algebraic algorithms and in particular GPCA in Section 3.1. First, as we mentioned this algorithm is not tractable in the dimension of the subspaces, meaning that a polynomial-time algorithm does not exist. Second, as we explained the sampling complexity of GPCA is not well understood. Another main issue is that GPCA is not robust to noise although some heuristics have been developed to address this issue, see e.g. [160]. An interesting approach to make GPCA robust is based on semidefinite programming [197]. However, this novel formulation is still intractable in the dimension of the subspaces and it is not clear how the performance of the algorithm depends upon the parameters of interest.

We also discussed iterative methods such as K-subspace algorithm [227], where the subspace clustering problem is formulated as a non-convex optimization problem over the choice of bases for each subspace as well as a set of variables indicating the correct segmentation. As detailed in Section 3.2 a cost function is then iteratively optimized over the basis and the segmentation variables. Each iteration is computationally tractable. However, due to the non-convex nature of the problem, the convergence of the sequence of iterates is only guaranteed to a local minimum. As a consequence, the dependence upon the key parameters is not well understood. Furthermore, the algorithm can be sensitive to noise and outliers. Other examples of iterative methods may be found in [6, 47, 157, 251]. In Section 3.3, we discussed statistical methods such as Mixtures of Probabilistic PCA (MPPCA) [223] and Agglomerative Lossy Compression (ALC) [158]. These methods are also non-convex in nature and suffer from the same issues as iterative methods.

Many other methods apply spectral clustering to a specially constructed graph [9, 15, 46, 72, 73, 116, 248, 252]. They share the same difficulties as stated above and [233] discusses advantages and drawbacks. One approach of this kind is termed Sparse Curvature Clustering (SCC) [14, 15, 72, 73]. As we discussed earlier this approach is not tractable in the dimension of the subspaces as it requires building a tensor with $N^{(d+2)}$ entries and involves computations with this tensor. Some theoretical

guarantees for this algorithm are given in [72] although its limits of performance and robustness to noise are not fully understood. An approach similar to SSC is called *low-rank representation* (LRR) [149]. The LRR algorithm is tractable but its robustness to noise and its dependence upon key parameters is not understood. The work in [147] formulates the robust subspace clustering problem as a non-convex geometric minimization problem over the Grassmannian. Because of the non-convexity, this formulation may not be tractable. On the positive side, this algorithm is provably robust and can accommodate noise levels up to $\mathcal{O}(1/(Ld^{3/2}))$. However, the density ρ required for favorable properties to hold is an unknown function of the dimensions of the subspaces (e.g. ρ could depend on d in a super polynomial fashion). Also, the bound on the noise level seems to decrease as the dimension d and number of subspaces L increases. In contrast, our theory requires $\rho \geq \rho^*$ where ρ^* is a fixed numerical constant. We would also like to mention [100] which establishes robustness to sparse outliers but with a dependence on the key parameters that is super-polynomial in the dimension of the subspaces demanding $\rho \geq C_0 d^{\log n}$. (Numerical simulations in [100] seem to indicate that ρ cannot be a constant.)

Chapter 6

Numerical experiments

In this section we perform synthetic and real data experiments that complement our theoretical results. We begin by explaining the metrics of performance we use for evaluating subspace clustering algorithms.

6.1 Error metrics

In addition to the false/true discovery measures already introduced we use four different metrics:

- *Feature detection error.* For each point \mathbf{y}_i , partition the optimal solution of SSC/RSC as

$$\boldsymbol{\beta}_i = \boldsymbol{\Gamma} \begin{bmatrix} \boldsymbol{\beta}_{i1} \\ \boldsymbol{\beta}_{i2} \\ \vdots \\ \boldsymbol{\beta}_{iL} \end{bmatrix}.$$

In this representation, $\boldsymbol{\Gamma}$ is our unknown permutation matrix and $\boldsymbol{\beta}_{i1}, \boldsymbol{\beta}_{i2}, \dots, \boldsymbol{\beta}_{iL}$ denote the coefficients corresponding to each of the L subspaces. Using N as the total number of points, the feature detection error is

$$\frac{1}{N} \sum_{i=1}^N \left(1 - \frac{\|\boldsymbol{\beta}_{i\ell_i}\|_{\ell_1}}{\|\boldsymbol{\beta}_i\|_{\ell_1}} \right) \quad (6.1.1)$$

in which ℓ_i is the subspace \mathbf{y}_i belongs to. The quantity between brackets in (6.1.1) measures how far we are from choosing all our neighbors in the same subspace; when the subspace detection property holds, this term is equal to 0 whereas it takes on the value 1 when all the points are chosen from the other subspaces.

- *Clustering error.* The clustering error is simply defined as

$$\frac{\# \text{ of misclassified points}}{\text{total } \# \text{ of points}}. \quad (6.1.2)$$

- *Error in estimating the number of subspaces.* This is a 0-1 error which takes on the value 0 if the true number of subspaces is correctly estimated, and 1 otherwise.
- *Smallest nonzero eigenvalue.* We use the $(N-L)+1$ -th smallest eigenvalue of the normalized Laplacian¹ as a numerical check on whether the subspace detection property holds (when the subspace detection property holds this value vanishes).

6.2 Synthetic experiments

This section proposes numerical experiments on synthesized data to further our understanding of the behavior/limitations of SSC, our different proposals for robust subspace clustering, and of our analysis of these approaches. Since we have performed multiple synthetic numerical studies for subspace clustering in the presence of noise in Section 5.4, here we focus on the noiseless, missing data and gross outlier cases.

¹After building the symmetrized affinity graph $\mathbf{W} = |\mathbf{Z}| + |\mathbf{Z}|^T$, we form the normalized Laplacian $\mathbf{L}_N = \mathbf{I} - \mathbf{D}^{-1/2} \mathbf{W} \mathbf{D}^{-1/2}$, where \mathbf{D} is a diagonal matrix and D_{ii} is equal to the sum of the elements in column \mathbf{W}_i . This form of the Laplacian works better for spectral clustering as observed in many applications [189].

6.2.1 Segmentation with noiseless data

As mentioned in Section 5.3.3, the subspace detection property can hold even when the dimensions of the subspaces are large in comparison with the ambient dimension n . SSC can also work beyond the region where the subspace detection property holds because of further spectral clustering. In Section 6.2.1.1, we demonstrate that the subspace detection property can hold even when the subspaces intersect. In Section 6.2.1.2, we study the performance of SSC under changes in the affinity between subspaces and the number of points per subspace. In Section 6.2.1.3, we illustrate the effect of the dimension of the subspaces on the subspace detection property and the spectral gap.

6.2.1.1 Subspace detection property holds even when the subspaces intersect

We wish to demonstrate that the subspace detection property holds even when the subspaces intersect. To this end, we generate two subspaces of dimension $d = 10$ in $\mathbb{R}^{n=200}$ with an intersection of dimension s . We sample one subspace (S_1) of dimension d uniformly at random among all d -dimensional subspaces and a subspace of dimension s (denoted by $S_2^{(1)}$) inside that subspace, again, uniformly at random. Sample another subspace $S_2^{(2)}$ of dimension $d - s$ uniformly at random and set $S_2 = S_2^{(1)} \oplus S_2^{(2)}$.

Our experiment selects $N_1 = N_2 = 20d$ points uniformly at random from each subspace. We generate 20 instances from this model and report the average of the first three error criteria over these instances, see Figure 6.1. Here, the subspace detection property holds up to $s = 3$. Also, after the spectral clustering step, SSC has a vanishing clustering error even when the dimension of the intersection is as large as $s = 6$.

6.2.1.2 Effect of the affinity between subspaces

In Section 5.3.2, we showed that in the semi-random model, the success of SSC depends upon the affinity between the subspaces and upon the density of points per subspace (recovery becomes harder as the affinity increases and as the density

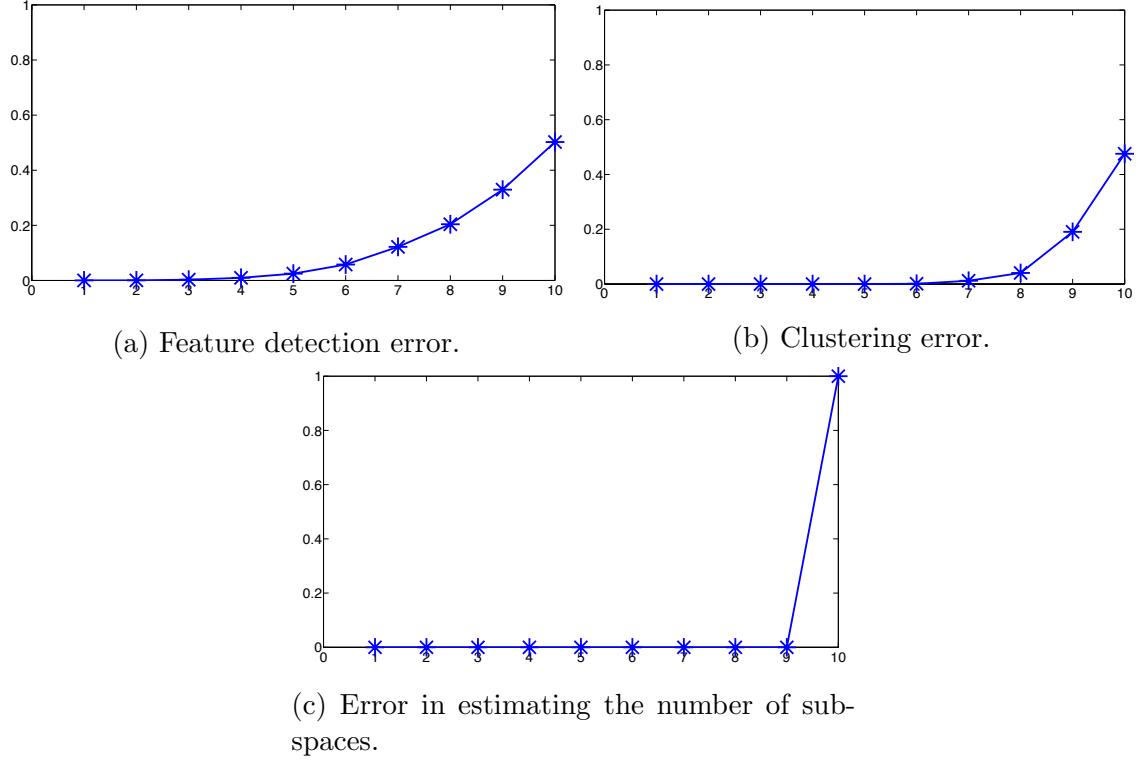


Figure 6.1: Error metrics as a function of the dimension of the intersection.

of points per subspace decreases). We study here this trade-off in greater details through experiments on synthetic data.

We generate 3 subspaces S_1 , S_2 , and S_3 , each of dimension $d = 20$ in $\mathbb{R}^{n=40}$. The choice $n = 2d$ makes the problem challenging since every data point on one subspace can also be expressed as a linear combination of points on other subspaces. The bases we choose for S_1 and S_2 are

$$\mathbf{U}^{(1)} = \begin{bmatrix} \mathbf{I}_d \\ \mathbf{0}_{d \times d} \end{bmatrix}, \quad \mathbf{U}^{(2)} = \begin{bmatrix} \mathbf{0}_{d \times d} \\ \mathbf{I}_d \end{bmatrix}, \quad (6.2.1)$$

whereas for S_3 ,

$$\mathbf{U}^{(3)} = \begin{bmatrix} \cos(\theta_1) & 0 & 0 & 0 & \dots & 0 \\ 0 & \cos(\theta_2) & 0 & 0 & \dots & 0 \\ 0 & 0 & \cos(\theta_3) & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \dots & \cos(\theta_d) \\ \sin(\theta_1) & 0 & 0 & 0 & \dots & 0 \\ 0 & \sin(\theta_2) & 0 & 0 & \dots & 0 \\ 0 & 0 & \sin(\theta_3) & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \dots & \sin(\theta_d) \end{bmatrix}. \quad (6.2.2)$$

Above, the principal angles are set in such a way that $\cos \theta_i$ decreases linearly from $\cos \theta$ to $\alpha \cos \theta$, where θ and α are fixed parameters; that is to say, $\cos \theta_i = (1 - a(i-1)) \cos \theta$, $a = \frac{1-\alpha}{d-1}$.

In our experiments we sample ρd points uniformly at random from each subspace. We fix $\alpha = \frac{1}{2}$ and vary $\rho \in [2, 10]$ and $\theta \in [0, \frac{\pi}{2}]$. Since $\alpha = \frac{1}{2}$, as θ increases from 0 to $\pi/2$, the maximum affinity $\max_{i \neq j} \text{aff}(S_i, S_j)$ decreases from 1 to 0.7094 (recall that a normalized affinity equal to 1 indicates a perfect overlap, i.e. two subspaces are the same). For each value of ρ and θ , we evaluate the SSC performance according to the three error criteria above. The results, shown in Figure 6.2, indicate that SSC is successful even for large values of the maximum affinity as long as the density is sufficiently large. Also, the figures display a clear correlation between the three different error criteria indicating that each could be used as a proxy for the other two. An interesting point is $\rho = 3.25$ and $\text{aff} = 0.9$; here, the algorithm can identify the number of subspaces correctly and perform perfect subspace clustering (clustering error is 0). This indicates that the SSC algorithm in its full generality can achieve perfect subspace clustering even when the subspaces are very close.

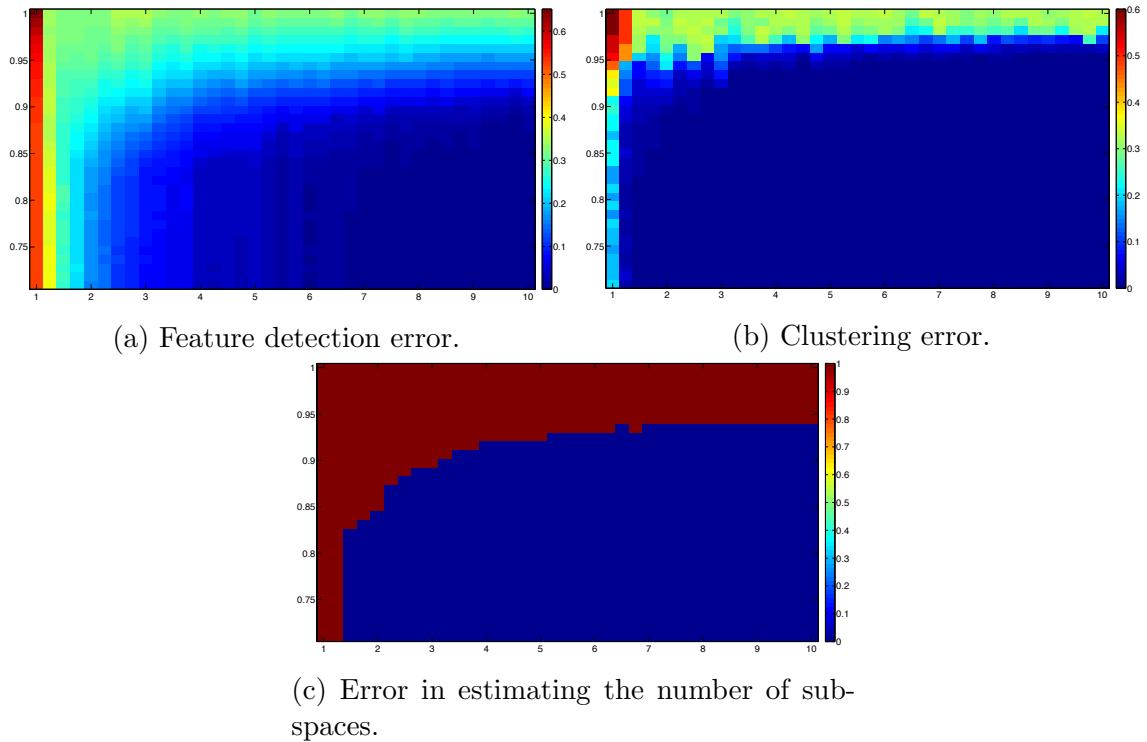


Figure 6.2: Performance of the SSC algorithm for different values of the affinity and density of points per subspace. In all three figures, the horizontal axis is the density ρ , and the vertical axis is the maximum affinity $\max_{i \neq j} \text{aff}(S_i, S_j)$.

6.2.1.3 Effect of dimension on subspace detection property and spectral gap

In order to illustrate the effect an increase in the dimension of subspaces has on the spectral gap, we generate $L = 20$ subspaces chosen uniformly at random from all d -dimensional subspaces in \mathbb{R}^{50} . We consider 5 different values for d , namely, 5, 10, 15, 20, 25. In all these cases, the total dimension of the subspaces Ld is more than the ambient dimension $n = 50$. We generate $4d$ unit-normed points on each subspace uniformly at random. The corresponding singular values of the normalized Laplacian are displayed in Figure 6.3. As evident from this figure, the subspace detection property holds, when the dimension of the subspaces is less than 10 (this corresponds to the last eigenvalues being exactly equal to 0). Beyond $d = 10$, the gap is still evident, however, the gap decreases as d increases. In all these cases, the gap was detectable using the sharpest descent heuristic presented in step 3 of Algorithm 4 and thus, the correct estimates for the number of subspaces were always found.

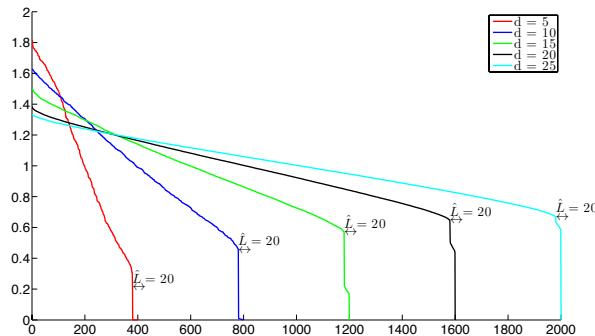


Figure 6.3: Gaps in the eigenvalues of the normalized Laplacian as a function of subspace dimension.

6.2.2 Segmentation with outliers

We now turn to outlier detection. For this purpose, we consider three different setups in which

- $d = 5, n = 50,$

- $d = 5, n = 100,$
- $d = 5, n = 200.$

In each case, we sample $L = 2n/d$ subspaces chosen uniformly at random so that the total dimension $Ld = 2n$. For each subspace, we generate $5d$ points uniformly at random so that the total number of data points is $N_d = 10n$. We add $N_0 = N_d$ outliers chosen uniformly at random on the sphere. Hence, the number of outliers is equal to the number of data points. The optimal values of the optimization problems (4.1.1) are plotted in Figure 6.4. The first N_d values correspond to the data points and the next N_0 values to the outliers. As can be seen in all the plots, a gap appears in the values of the ℓ_1 norm of the optimal solutions. That is, the optimal value for data points is much smaller than the corresponding optimal value for outlier points. We have argued that the critical parameter for outlier detection is the ratio d/n . The smaller, the better. As can be seen in Figure 6.4 (a), The ratio $d/n = 1/10$ is already small enough for the conjectured threshold of Algorithm 6 to work and detect all outlier points correctly. However, it wrongfully considers a few data points as outliers. In Figure 6.4 (b), $d/n = 1/20$ and the conjectured threshold already works perfectly but the proven threshold is still not able to do outlier detection well. In Figure 6.4 (c), $d/n = 1/40$, both the conjectured and proven thresholds can perform perfect outlier detection. (In practice it is of course not necessary to use the threshold as a criterion for outlier detection; one can instead use a gap in the optimal values.) It is also worth mentioning that if d is larger, the optimal value is more concentrated for the data points and, therefore, both the proven and conjectured threshold would work for smaller ratios of d/n (this is different from the small values of d above).

6.2.3 Segmentation with missing data

We now turn our attention to synthetic experiments with missing data. In order to be able to compare with a previous high rank matrix completion approach [100] we will follow the same experimental setup and choose $n = 100$, $N = 5000$, $L = 10$, and $d = 5$. We picked the L subspaces of dimension d uniformly at random from the space of d dimensional subspaces in \mathbb{R}^n . As a result the incoherence assumption is satisfied. For

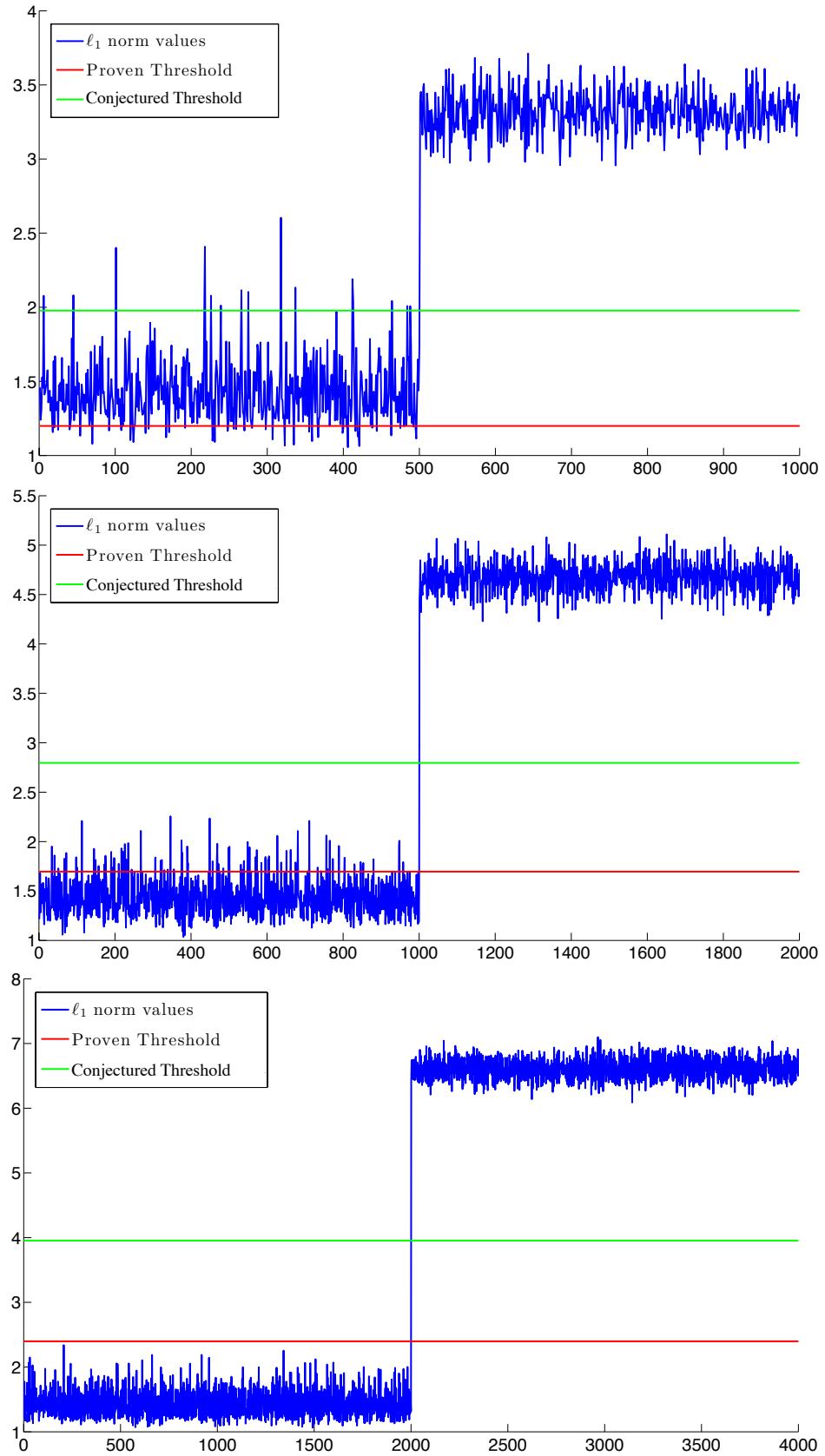


Figure 6.4: Gap in the optimal values with $L = 2n/d$ subspaces. (a) $d = 5, n = 50, L = 20$. (b) $d = 5, n = 100, L = 40$. (c) $d = 5, n = 200, L = 80$.

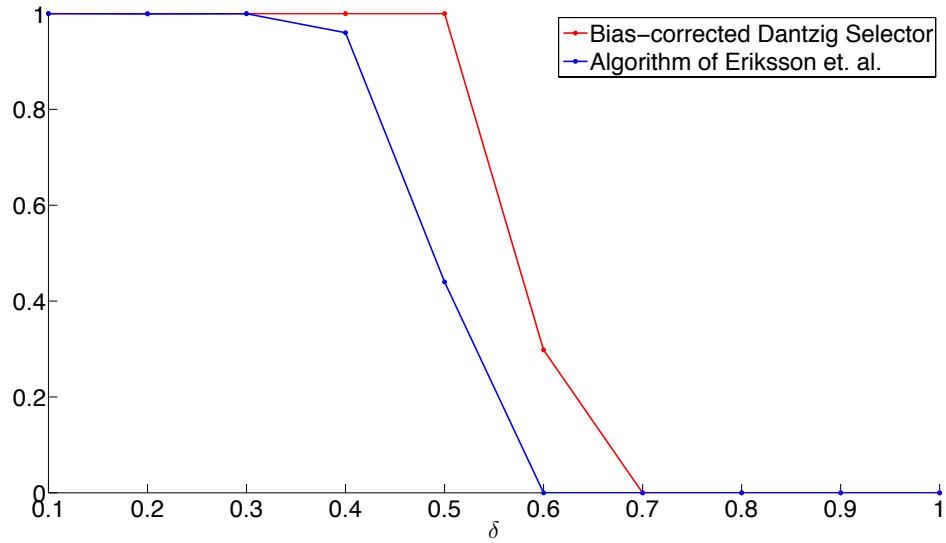


Figure 6.5: The fraction of correctly completed columns (with a tolerance of 10^{-5}), versus the fraction of missing entries δ for the bias-corrected Dantzig Selector and the algorithm suggested in [100].

each subspace, we generate 500 points drawn from a $\mathcal{N}(0, \mathbf{U}\mathbf{U}^T)$ distribution, where $\mathbf{U} \in \mathbb{R}^{n \times d}$ is an orthonormal basis for that subspace. For the clustering step we used the bias corrected Dantzig selector with the normalization steps suggested in Section 4.6.1 and in particular with the choice of $\lambda = \frac{\sqrt{2\log N}}{\sqrt{n}} \sqrt{\frac{\bar{\delta}}{1-\bar{\delta}}}$. After identifying the clusters we used OPTSPACE [141] to complete the matrix associated with each cluster. We ran 50 independent trials of our procedure and compared it to the procedure reported in [100]. The results are summarized in Figure 6.5. This figure indicates that RSC-M can handle a higher fraction of missing entries. We note that in our procedure we did not have to set any parameters, while the algorithm of [100] require tuning of 5 different parameters (Please see Algorithm 1 in [100] for further details).

6.3 Experiments on temporal segmentation of motion capture data

In this section, we perform numerical experiments corroborating our main results and suggesting their applications to temporal segmentation of motion capture data. As we explained before, in this application we are given sensor measurements at multiple joints of the human body captured at different time instants. The goal is to segment the sensory data so that each cluster corresponds to the same activity.

We use the Carnegie Mellon Motion Capture dataset (available at <http://mocap.cs.cmu.edu>), which contains 149 subjects performing several activities (data are provided in [253]). The motion capture system uses 42 markers per subject. We consider the data from subject 86 in the dataset, consisting of 15 different trials, where each trial comprises multiple activities. We use trials 2 and 5, which feature more activities (8 activities for trial 2 and 7 activities for trial 5) and are, therefore, harder examples relative to the other trials.

We compare three different algorithms: a baseline algorithm, the two-step procedure and the bias-corrected Dantzig selector. We evaluate these algorithms based on the *clustering error*. That is, we assume knowledge of the number of subspaces and apply spectral clustering to the similarity matrix built by the algorithm. After the spectral clustering step, the clustering error is simply the ratio of misclassified points to the total number of points. We report our results on half of the examples—downsampling the video by a factor 2 keeping every other frame—as to make the problem more challenging. (As a side note, it is always desirable to have methods that work well on a smaller number of examples as one can use split-sample strategies for tuning purposes).²

As a baseline for comparison, we apply spectral clustering to a standard similarity graph built by connecting each data point to its K -nearest neighbors. For pairs of data points, \mathbf{y}_i and \mathbf{y}_j , that are connected in the K -nearest neighbor graph, we define the similarities between them by $W_{ij} = \exp(-\|\mathbf{y}_i - \mathbf{y}_j\|_2^2/t)$, where $t > 0$ is a

²We have adopted this subsampling strategy to make our experiments reproducible. For tuning purposes, a random strategy may be preferable.

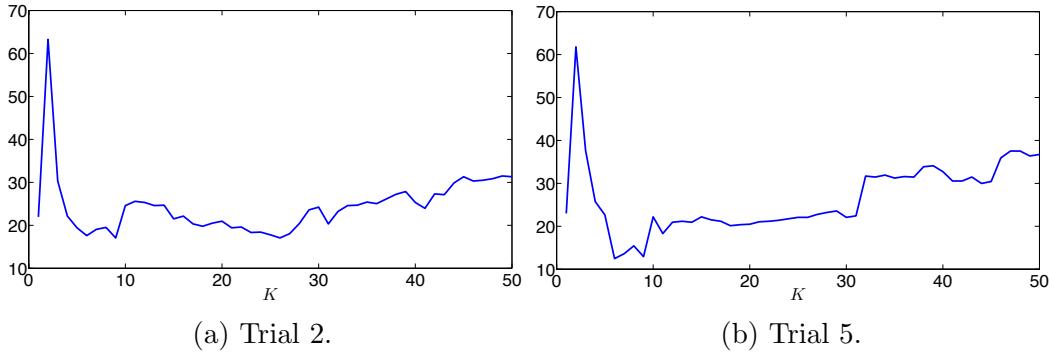


Figure 6.6: Minimum clustering error (%) for each K in the baseline algorithm.

tuning parameter (a.k.a. temperature). For pairs of data points, \mathbf{y}_i and \mathbf{y}_j , that are not connected in the K -nearest neighbor graph, we set $W_{ij} = 0$. Thus, pairs of neighboring data points that have small Euclidean distances from each other are considered to be more similar, since they have high similarity W_{ij} . We then apply spectral clustering to the similarity graph and measure the clustering error. For each value of K , we record the minimum clustering error over different choices of the temperature parameter $t > 0$ as shown in Figures 6.6a and 6.6b. The minimum clustering error for trials 2 and 5 are 17.06% and 12.47%.

For solving the LASSO problems in RSC-N, we developed a computational routine made publicly available [1] based on TFOCS [35] solving the optimization problems in parallel. For the corrected Dantzig selector of RSC-D we use a homotopy solver in the spirit of [229].

For both RSC-N and RSC-D we normalize the data points as a preprocessing step. We work with a noise σ in the interval $[0.001, 0.045]$, and use $f(t) = \alpha/t$ with values of α around $1/4$ (this is equivalent to varying λ around $1/\lambda_o = 4 \|\boldsymbol{\beta}^*\|_{\ell_1}$) in RSC-N. For RSC-D, we vary λ around $\lambda_o = \sqrt{2/n} \sigma \sqrt{1 + \sigma^2}$. After building the similarity graph from the sparse regression output, we apply spectral clustering as explained earlier. Figures 6.7a, 6.7b, 6.8a, and 6.8b show the clustering error (on trial 5) and the red point indicates the location where the minimum clustering error is reached. Figures 6.7a and 6.7b show that for RSC-N the value of the clustering error is not overly sensitive to the choice of σ —especially around $\lambda = \lambda_o$. Notice that the clustering error for RSC-N and RSC-D are significantly lower than the baseline algorithm for a

	Baseline Algorithm	RSC-N	RSC-D
Trial 2	17.06%	3.54%	9.53%
Trial 5	12.47%	4.35%	4.92%

Table 6.1: Minimum clustering error.

wide range of parameter values. The reason the baseline algorithm performs poorly in this case is that there are many points that are in small Euclidean distances from each other, but belong to different subspaces.

Finally a summary of the clustering errors of these algorithms on the two trials are reported in Table 1. Our robust subspace clustering techniques outperform the baseline algorithm. This shows that the multiple subspace model is better for clustering purposes. RSC-N seems to work slightly better than RSC-D for these two examples. Table 2 reports the optimal parameters that achieve the minimum clustering error for each algorithm. The table indicates that on real data, choosing λ close to λ_o also works very well. Also, one can see that in comparison with the synthetic simulations of Section 4.4.2, a more conservative choice of the regularization parameter λ is needed for the corrected Dantzig selector as λ needs to be chosen much higher than λ_o to achieve the best results. This may be attributed to the fact that the subspaces in this example are very close to each other and are not drawn at random as was the case with our synthetic data. To get a sense of the affinity values, we fit a subspace of dimension d_ℓ to the N_ℓ data points from the ℓ th group, where d_ℓ is chosen as the smallest nonnegative integer such that the partial sum of the d_ℓ top singular values is at least 90% of the total sum. Figure 6.9 shows that the affinities are higher than 0.75 for both trials.

	Baseline algorithm	RSC-N	RSC-D
Trial 2	$K=9, t=0.0769$	$\sigma = 0.03, \lambda = 1.25\lambda_o$	$\sigma = 0.004, \lambda = 41.5\lambda_o$
Trial 5	$K=6, t=0.0455$	$\sigma = 0.01, \lambda = \lambda_o$	$\sigma = 0.03, \lambda = 45.5\lambda_o$

Table 6.2: Optimal parameters.

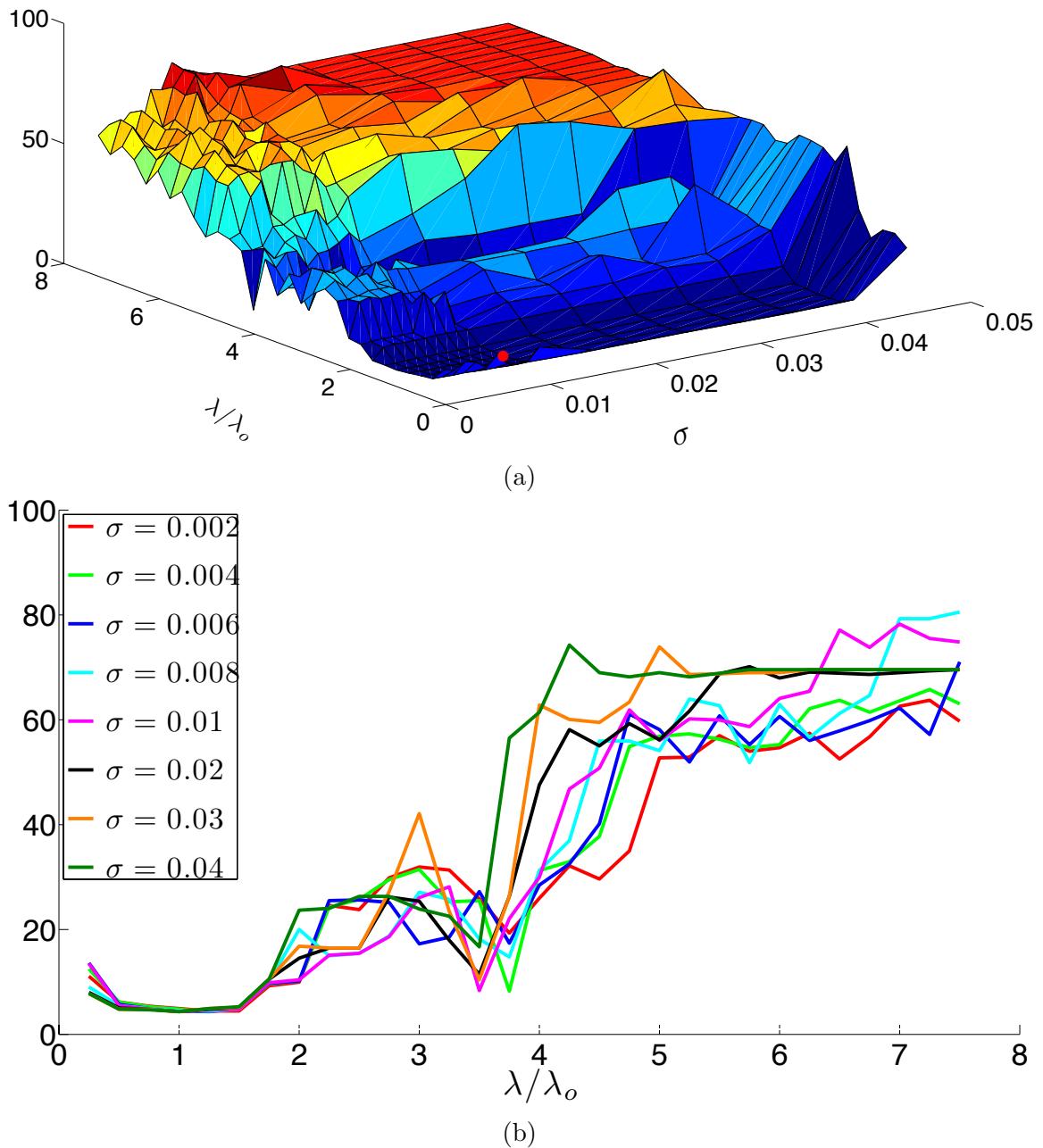


Figure 6.7: Clustering error (%) for different values of λ and σ on trial 5 using RSC-N
 (a) 3D plot (minimum clustering error appears in red). (b) 2D cross sections.

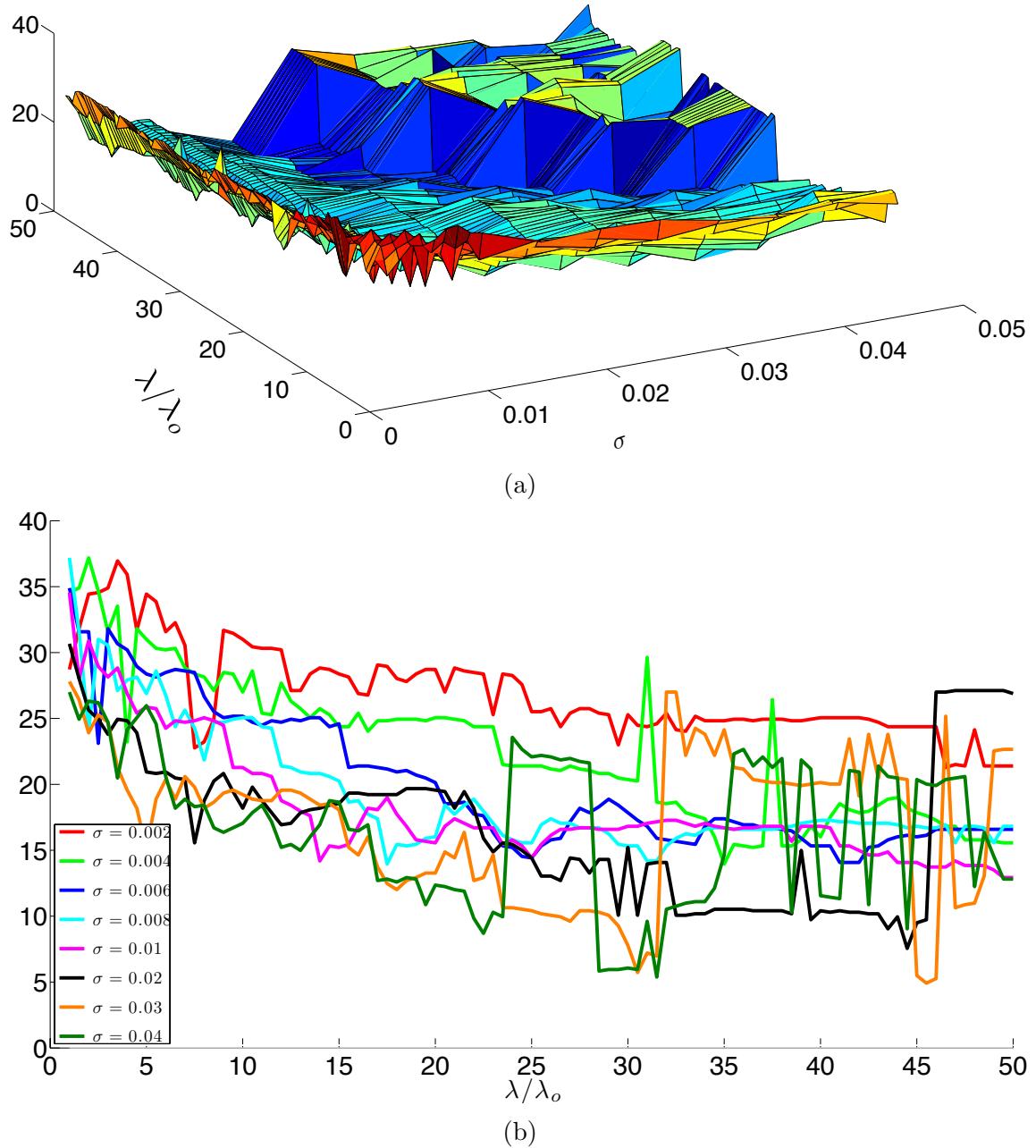


Figure 6.8: Clustering error (%) for different values of λ and σ on trial 5 using RSC-D. (a) 3D plot (minimum clustering error appears in red). (b) 2D cross sections.

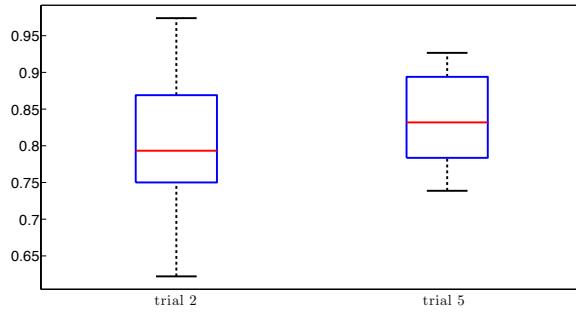


Figure 6.9: Box plot of the affinities between subspaces for trials 2 and 5.

6.4 Experiments on motion segmentation data

In this section we compare the performance of Robust Subspace Clustering (RSC) with some other subspace clustering algorithms: LSA [247], SCC [73], LRR [149], LRSC [101], and cosine-based affinity (KNN-angle) [127, 128, 145]. These simulations are a recreation of the results in [96, 97] with some additions. We use RSC-N but use the same value of λ for all data points. That is, we solve the following series of optimization problems

$$\frac{1}{2} \|\mathbf{y}_i - \mathbf{Y}\boldsymbol{\beta}\|_{\ell_2}^2 + \lambda \|\boldsymbol{\beta}\|_{\ell_1} \quad \text{subject to} \quad \boldsymbol{\beta}_i = 0 \quad \text{for } i = 1, 2, \dots, N,$$

where λ is fixed for all data points. As discussed in our theoretical results this is a good choice when we know the subspaces are of equal dimension which is the case in motion segmentation. We refer to this algorithm as RSC-L. We use the best choice of λ in all our experiments. For the other algorithms we use the code provided by each of the authors online and we follow the same implementation details as [97].

We run our experiments on the Hopkins 155 dataset [225]. This dataset contains 155 video sequences of 2 or 3 motions (corresponding to 2 or 3 lower dimensional subspaces). 120 of these videos have two motions and 35 have three motions. The result of applying subspace clustering algorithms to the dataset is shown in Table 6.3. This table shows that RSC-L outperforms other algorithms.

	LSA	SCC	KNN-angle	LRR	LRSC	RSC-L
2 Motions	4.23	2.89	18.83	4.10	3.69	1.52
3 Motions	7.02	8.25	29.26	9.89	7.69	4.40
All	4.86	4.10	21.18	4.10	4.59	2.18

Table 6.3: Clustering error (%) of different algorithms on the Hopkins 155 dataset.

	LSA	SCC	KNN-angle	LRR	LRSC	RSC-S
2 subjects	32.80	16.62	14.61	9.52	5.32	1.86
3 subjects	52.29	38.16	21.78	19.52	8.47	3.10
5 subjects	58.02	58.90	28.92	34.16	12.24	4.31
8 subjects	59.19	66.11	35.03	41.19	23.72	5.85
10 subjects	60.42	73.02	38.07	38.85	30.36	10.94

Table 6.4: Clustering error (%) of different algorithms on the Extended Yale B dataset.

6.5 Experiments on face clustering

In this section we compare the performance of RSC-S with the other algorithms mentioned in the previous section on face clustering data that contains sparse outlying entries. We run our experiments on the Extended Yale B data set [74]. This dataset contains pictures of size 192×168 pixels from 38 individuals. For each individual there is 64 pictures under various lightning conditions. As in [96] we down sample these images to size 48×42 to reduce the computational cost of the algorithms. We choose random combinations of subjects of sizes 2, 3, 5, 8, and 10 according to the scheme described in Section 7.2 of [96]. We apply the different algorithms on this data set. The results are reported in Table 6.4. These results indicate that RSC-S is very good at dealing with sparse outliers.

6.6 Experiments on cancer data

We use the data set in preprocessed form from the [MIT Broad Institute](#) which contains various gene expression data sets. In each data set different categories correspond to

different tumor or tissue types relating to different cancers. We tested our results on four data sets: Leukemia, St. Jude leukemia, Lung cancer and Novartis multi-tissue. A summary of the data is provided in Table 6.5.

	number of clusters	number of biological samples	number of patients
Leukemia	3	999	38
St. Jude leukemia	6	985	248
Lung cancer	4	1000	197
Novartis multi-tissue	4	1000	103

Table 6.5: Summary of the data.

6.6.1 No missing entries

We first perform some clustering experiments when there are no missing entries. We compare three different algorithms: KNN (K Nearest Neighbors), Kmeans and RSC-L. For both RSC-L and KNN we assume that the columns are normalized to have Euclidean norm 1. This is the case where the best results are obtained. The performance of Kmeans is not affected by this normalization so we do not apply it. We evaluate these algorithms based on the clustering error.

For comparison, we apply spectral clustering to a standard similarity graph built by connecting each data point to its K -nearest neighbors. For pairs of data points, \mathbf{y}_i and \mathbf{y}_j , that are connected in the K -nearest neighbor graph, we define the similarities between them by $W_{ij} = \exp(-\|\mathbf{y}_i - \mathbf{y}_j\|_2^2/t)$, where $t > 0$ is a tuning parameter. For pairs of data points, \mathbf{y}_i and \mathbf{y}_j , that are not connected in the K -nearest neighbor graph, we set $W_{ij} = 0$. We then apply spectral clustering to the similarity graph and measure the clustering error.

The best results obtained by different algorithms are reported in Table 6.6. Except for the Leukemia data set, RSC-L outperforms the other two methods. Note that in the Leukemia data set we have very few patients with 3 different categories of sizes (19,8,11) which suggests that we do not have enough samples for the multi-subspace assumption to hold. Furthermore, an error of 2.63% and 5.26% for Kmeans and

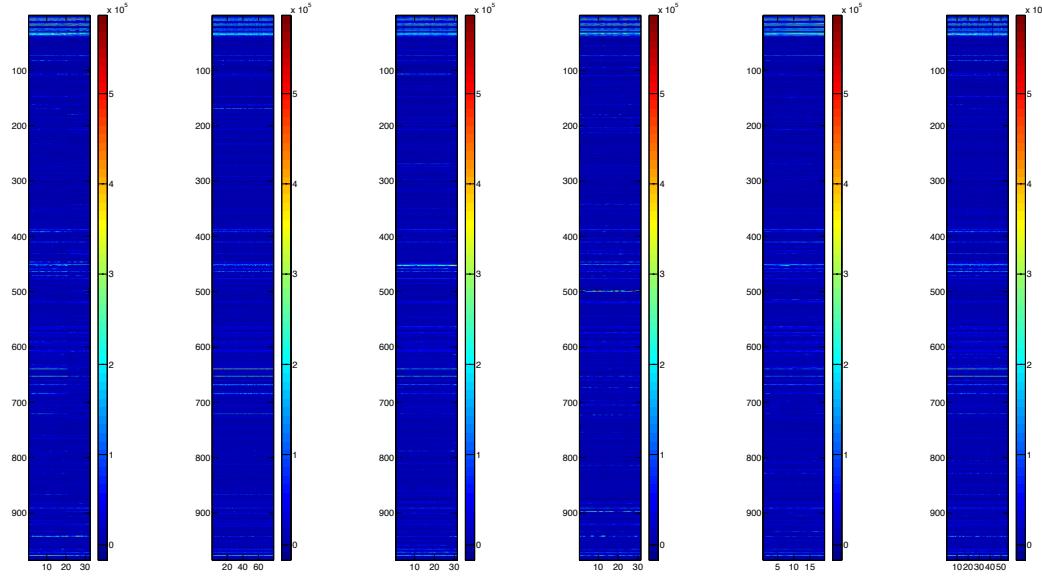


Figure 6.10: Heat map plot of the gene expression level of the different groups of patients in the St. Jude Leukemia data set.

RSC-L correspond to 1/38 and 2/38 mistakes which is not a substantial difference with 0/38 for KNN.

	KNN	Kmeans	RSC-L
Leukemia	0%	2.63%	5.26%
St. Jude leukemia	11.69%	17.34%	9.68%
Lung cancer	15.74%	30.96%	5.58%
Novartis multi-tissue	6.80%	35.92%	2.91%

Table 6.6: Minimum clustering error for various algorithms and data sets.

Finally, to show the importance of clustering for this sort of problem we depict the gene expression data for St. Jude leukemia in Figure 6.10. Note that in this figure there are certain visible lines in each category that do not exist in other categories. These lines correspond to genes (rows). Thus, this figure shows which genes (rows) are more important in each category.

6.6.2 With missing entries

In this section we test the performance of RSC-M when some of the entries in the data matrix are missing. For this purpose we use the same cancer data set. We randomly delete each entry of the matrix independent from others with probability δ and try to cluster the resulting matrix. For each data matrix we generate 10 different sampling patterns and record the average clustering error over these patterns. As a baseline we compare our method with kmeans. The results are depicted in Figure 6.11. The plots indicate that the algorithm is effective, especially at high fraction of missing entries.

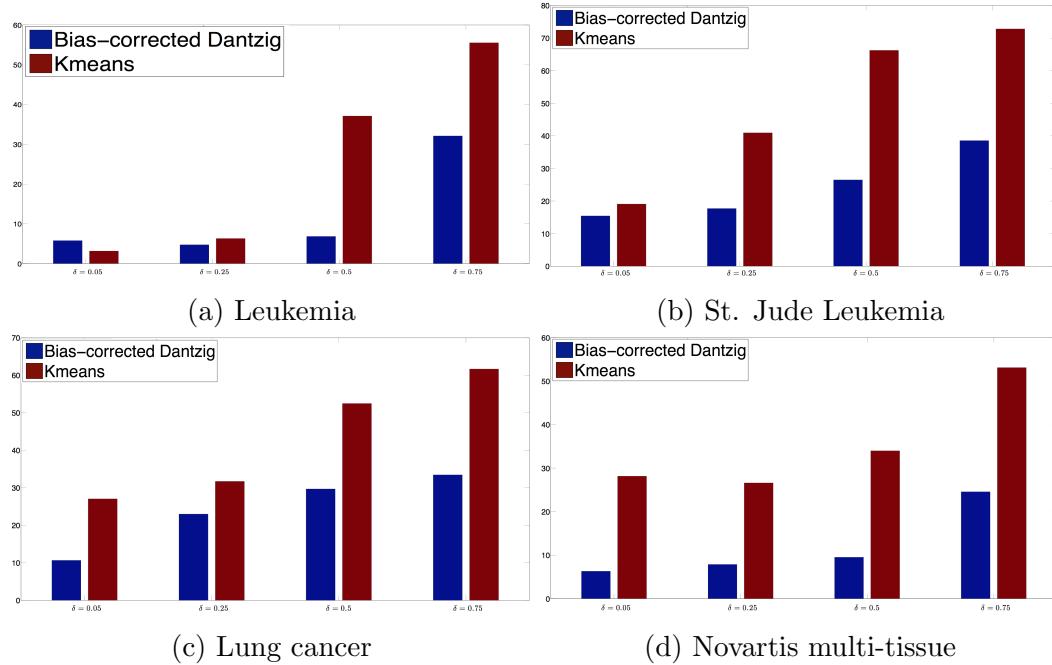


Figure 6.11: Clustering error (%) for different cancer data examples and different fractions of missing entries δ .

6.7 Experiments on Flicker photos of animals

We now explore the use of our subspace clustering techniques in semi-supervised learning applications. There is an ever growing number of images on the web and automatic labeling of these images is among the most challenging unsupervised learning problems. We perform some preliminary experiments on Flicker photos of animals from NUS-WIDE database [76]. Given these images of the animals the goal is to cluster them based on animal type. NUS-WIDE data base a large collection of Flicker images of animals. Some of these pictures are depicted in Figure 6.12. As can be seen these pictures contains animals under different pose, background, illumination and even sometimes multiple animals per frame. As a result this dataset is extremely challenging for clustering purposes.



Figure 6.12: Sample Flickr images from NUS-WIDE database.

From these images we extract five different low-level features, namely 64-D color histogram, 144-D color correlogram, 73-D edge direction histogram, 128-D wavelet texture, 225-D block wise color moments. The concatenation of these features makes

up a single data point for our subspace clustering problem. We perform a variation of RSC with the following regression problem

$$\|\mathbf{y}_i - \mathbf{Y}\boldsymbol{\beta}\|_{\ell_1} + \lambda \|\boldsymbol{\beta}\|_{\ell_1} \quad \text{subject to} \quad \mathbf{1}^T \boldsymbol{\beta} = 1 \quad \text{and} \quad \boldsymbol{\beta}_i = 0 \quad \text{for } i = 1, 2, \dots, N.$$

We refer to this variation as RSC-aff-S. We note that this variation of RSC is a combination of the affine and sparse outlying variations. In Figure 6.13 we report clustering errors on images of pairs of animals in this data set which greatly improves on the KNN and kmeans schemes. While these results are promising we caution that more experiments are required to test the effectiveness of our approach for this particular problem. Furthermore, we would like to emphasize that we are not claiming that this problem an instance of a subspace clustering problem. We are only suggesting that sparse regression using self representation can be effective in other settings.

	KNN	kmeans	RSC-aff-S
 \leftrightarrow 	%20.51	%28.21	%15.71
 \leftrightarrow 	%18.73	%33.71	% 3.37
 \leftrightarrow 	%15.38	%15.93	% 9.89

Figure 6.13: Clustering errors on images of pairs of animals in NUS-WIDE dataset.

Chapter 7

Proofs

In this chapter we shall prove all of the results presented in Chapter 5. We first prove the results corresponding to noiseless data before moving on to results on noise and missing data. Before we begin we introduce concepts such as \mathcal{K} -norms and polar sets, which will play a crucial role in our analysis.

7.1 Linear programming theory

We are interested in finding the support of the optimal solution to

$$\min_{\mathbf{x} \in \mathbb{R}^N} \|\mathbf{x}\|_{\ell_1} \quad \text{subject to} \quad \mathbf{A}\mathbf{x} = \mathbf{y}, \quad (7.1.1)$$

where both \mathbf{y} and the columns of \mathbf{A} have unit norm. The dual takes the form

$$\max_{\mathbf{z} \in \mathbb{R}^n} \langle \mathbf{y}, \mathbf{z} \rangle \quad \text{subject to} \quad \|\mathbf{A}^T \mathbf{z}\|_{\ell_\infty} \leq 1. \quad (7.1.2)$$

Since strong duality always holds in linear programming, the optimal values of (7.1.1) and (7.1.2) are equal. We now introduce some notation to express the dual program differently.

Definition 7.1.1 *The norm of a vector \mathbf{y} with respect to a symmetric convex body*

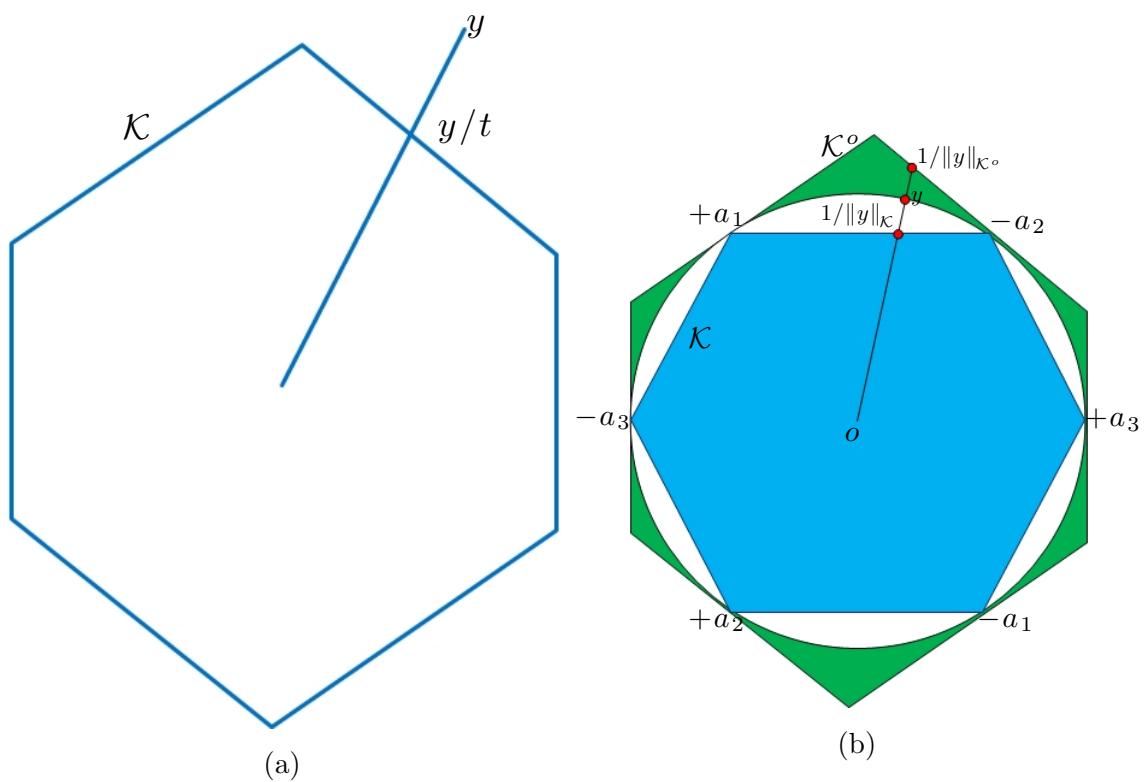


Figure 7.1: Illustration of Definitions 7.1.1 and 7.1.2. (a) Norm with respect to a polytope \mathcal{K} . (b) Polytope \mathcal{K} and its polar \mathcal{K}^o .

is defined as

$$\|\mathbf{y}\|_{\mathcal{K}} = \inf \{t > 0 : \mathbf{y}/t \in \mathcal{K}\}. \quad (7.1.3)$$

This norm is shown in Figure 7.1a.

Definition 7.1.2 *The polar set \mathcal{K}^o of $\mathcal{K} \subset \mathbb{R}^n$ is defined as*

$$\mathcal{K}^o = \{\mathbf{y} \in \mathbb{R}^n : \langle \mathbf{x}, \mathbf{y} \rangle \leq 1 \text{ for all } \mathbf{x} \in \mathcal{K}\}. \quad (7.1.4)$$

Set $\mathcal{K}^o = \{\mathbf{z} : \|\mathbf{A}^T \mathbf{z}\|_{\ell_\infty} \leq 1\}$ so that our dual problem (7.1.2) is of the form

$$\max_{\mathbf{z} \in \mathbb{R}^n} \langle \mathbf{y}, \mathbf{z} \rangle \quad \text{subject to} \quad \mathbf{z} \in \mathcal{K}^o. \quad (7.1.5)$$

It then follows from the definitions above that the optimal value of (7.1.1) is given by $\|\mathbf{y}\|_{\mathcal{K}}$, where $\mathcal{K} = \text{conv}(\pm \mathbf{a}_1, \dots, \pm \mathbf{a}_N)$; that is to say, the minimum value of the ℓ_1 norm is the norm of \mathbf{y} with respect to the symmetrized convex hull of the columns of \mathbf{A} . In other words, this perspective asserts that support detection in an ℓ_1 minimization problem is equivalent to finding the face of the polytope \mathcal{K} that passes through the ray $\vec{y} = \{t\mathbf{y}, t \geq 0\}$; the extreme points of this face reveal those indices with a nonzero entry. We will refer to the face passing through the ray \vec{y} as the face closest to \mathbf{y} . Figure 7.1b illustrates some of these concepts.

7.2 Proofs for noiseless data

To avoid repetition, we define the primal optimization problem $P(\mathbf{y}, \mathbf{A})$ as

$$\min_{\mathbf{x}} \|\mathbf{x}\|_{\ell_1} \quad \text{subject to} \quad \mathbf{A}\mathbf{x} = \mathbf{y},$$

and its dual $D(\mathbf{y}, \mathbf{A})$ as

$$\max_{\boldsymbol{\nu}} \langle \mathbf{y}, \boldsymbol{\nu} \rangle \quad \text{subject to} \quad \|\mathbf{A}^T \boldsymbol{\nu}\|_{\ell_\infty} \leq 1.$$

We denote the optimal solutions by $\text{optsolP}(\mathbf{y}, \mathbf{A})$ and $\text{optsold}(\mathbf{y}, \mathbf{A})$. Since the primal is a linear program, strong duality holds, and both the primal and dual have the same optimal value which we denote by $\text{optval}(\mathbf{y}, \mathbf{A})$ (the optimal value is set to infinity when the primal problem is infeasible). Also notice that as we discussed in Section 7.1 this optimal value is equal to $\|\mathbf{y}\|_{\mathcal{K}}$, where $\mathcal{K}(\mathbf{A}) = \text{conv}(\pm \mathbf{a}_1, \dots, \pm \mathbf{a}_N)$ and $\mathcal{K}^o(\mathbf{A}) = \{\mathbf{z} : \|\mathbf{A}^T \mathbf{z}\| \leq 1\}$.

7.2.1 Proof of Theorem 5.3.5

We first prove that the geometric condition (5.3.1) implies the subspace detection property. We begin by establishing a simple variant of a now classical lemma (e.g. see [63]). Below, we use the notation \mathbf{A}_S to denote the submatrix of \mathbf{A} with the same rows as \mathbf{A} and columns with indices in $S \subset \{1, \dots, N\}$.

Lemma 7.2.1 *Consider a vector $\mathbf{y} \in \mathbb{R}^n$ and a matrix $\mathbf{A} \in \mathbb{R}^{n \times N}$. If there exists \mathbf{c} obeying $\mathbf{y} = \mathbf{A}\mathbf{c}$ with support $S \subseteq T$, and a dual certificate vector $\boldsymbol{\nu}$ satisfying*

$$\mathbf{A}_S^T \boldsymbol{\nu} = \text{sgn}(\mathbf{c}_S), \quad \|\mathbf{A}_{T \cap S^c}^T \boldsymbol{\nu}\|_{\ell_\infty} \leq 1, \quad \|\mathbf{A}_{T^c}^T \boldsymbol{\nu}\|_{\ell_\infty} < 1,$$

then all optimal solutions \mathbf{z}^ to $P(\mathbf{y}, \mathbf{A})$ obey $\mathbf{z}_{T^c}^* = \mathbf{0}$.*

Proof Observe that for any optimal solution \mathbf{z}^* of $P(\mathbf{y}, \mathbf{A})$, we have

$$\begin{aligned} \|\mathbf{z}^*\|_{\ell_1} &= \|\mathbf{z}_S^*\|_{\ell_1} + \|\mathbf{z}_{T \cap S^c}^*\|_{\ell_1} + \|\mathbf{z}_{T^c}^*\|_{\ell_1} \\ &\geq \|\mathbf{c}_S\|_{\ell_1} + \langle \text{sgn}(\mathbf{c}_S), \mathbf{z}_S^* - \mathbf{c}_S \rangle + \|\mathbf{z}_{T \cap S^c}^*\|_{\ell_1} + \|\mathbf{z}_{T^c}^*\|_{\ell_1} \\ &= \|\mathbf{c}_S\|_{\ell_1} + \langle \boldsymbol{\nu}, \mathbf{A}_S(\mathbf{z}_S^* - \mathbf{c}_S) \rangle + \|\mathbf{z}_{T \cap S^c}^*\|_{\ell_1} + \|\mathbf{z}_{T^c}^*\|_{\ell_1} \\ &= \|\mathbf{c}_S\|_{\ell_1} + \|\mathbf{z}_{T \cap S^c}^*\|_{\ell_1} - \langle \boldsymbol{\nu}, \mathbf{A}_{T \cap S^c} \mathbf{z}_{T \cap S^c} \rangle + \|\mathbf{z}_{T^c}^*\|_{\ell_1} - \langle \boldsymbol{\nu}, \mathbf{A}_{T^c} \mathbf{z}_{T^c}^* \rangle. \end{aligned}$$

Now note that

$$\langle \boldsymbol{\nu}, \mathbf{A}_{T \cap S^c} \mathbf{z}_{T \cap S^c}^* \rangle = \langle \mathbf{A}_{T \cap S^c}^T \boldsymbol{\nu}, \mathbf{z}_{T \cap S^c}^* \rangle \leq \|\mathbf{A}_{T \cap S^c}^T \boldsymbol{\nu}\|_{\ell_\infty} \|\mathbf{z}_{T \cap S^c}^*\|_{\ell_1} \leq \|\mathbf{z}_{T \cap S^c}^*\|_{\ell_1}.$$

In a similar manner, we have $\langle \boldsymbol{\nu}, \mathbf{A}_{T^c} \mathbf{z}_{T^c}^* \rangle \leq \|\mathbf{A}_{T^c}^T \boldsymbol{\nu}\|_{\ell_\infty} \|\mathbf{z}_{T^c}^*\|_{\ell_1}$. Hence, using these two identities we get

$$\|\mathbf{z}^*\|_{\ell_1} \geq \|\mathbf{c}\|_{\ell_1} + (1 - \|\mathbf{A}_{T^c}^T \boldsymbol{\nu}\|_{\ell_\infty}) \|\mathbf{z}_{T^c}^*\|_{\ell_1}.$$

Since \mathbf{z}^* is an optimal solution, $\|\mathbf{z}^*\|_{\ell_1} \leq \|\mathbf{c}\|_{\ell_1}$, and plugging this into the last identity gives

$$(1 - \|\mathbf{A}_{T^c}^T \boldsymbol{\nu}\|_{\ell_\infty}) \|\mathbf{z}_{T^c}^{*T}\|_{\ell_1} \leq 0.$$

Now since $\|\mathbf{A}_{T^c}^T \boldsymbol{\nu}\|_{\ell_\infty} < 1$, it follows that $\|\mathbf{z}_{T^c}^*\|_{\ell_1} = 0$. ■

Consider $\mathbf{x}_i^{(\ell)} = \mathbf{U}^{(\ell)} \mathbf{a}_i^{(\ell)}$, where $\mathbf{U}^{(\ell)} \in \mathbb{R}^{n \times d_\ell}$ is an orthogonal basis for S_ℓ and define

$$\mathbf{c}_i^{(\ell)} = \text{optsolP}(\mathbf{a}_i^{(\ell)}, \mathbf{A}_{(-i)}^{(\ell)}).$$

Letting S be the support of $\mathbf{c}_i^{(\ell)}$, define $\boldsymbol{\vartheta}_i^{(\ell)}$ as an optimal solution to

$$\boldsymbol{\vartheta}_i^{(\ell)} = \arg \min_{\boldsymbol{\vartheta}_i^{(\ell)} \in \mathbb{R}^{d_\ell}} \|\boldsymbol{\vartheta}_i^{(\ell)}\|_{\ell_2} \quad \text{subject to} \quad \left\{ (\mathbf{A}_{(-i)}^{(\ell)})_S^T \boldsymbol{\vartheta}_i^{(\ell)} = \text{sgn}(\mathbf{c}_i^{(\ell)}), \left\| (\mathbf{A}_{(-i)}^{(\ell)})_{S^c}^T \boldsymbol{\vartheta}_i^{(\ell)} \right\|_{\ell_\infty} \leq 1 \right\}.$$

Because $\mathbf{c}_i^{(\ell)}$ is optimal for the primal problem, the dual problem is feasible by strong duality and the set above is nonempty. Also, $\boldsymbol{\vartheta}_i^{(\ell)}$ is a dual point in the sense of Definition 5.3.1; i.e. $\boldsymbol{\vartheta}_i^{(\ell)} = \boldsymbol{\vartheta}(\mathbf{a}_i^{(\ell)}, \mathbf{A}_{(-i)}^{(\ell)})$. Introduce

$$\boldsymbol{\nu}_i^{(\ell)} = \mathbf{U}^{(\ell)} \boldsymbol{\vartheta}_i^{(\ell)},$$

so that the direction of $\boldsymbol{\nu}_i^{(\ell)}$ is the i th dual direction; i.e. $\boldsymbol{\nu}_i^{(\ell)} = \|\boldsymbol{\vartheta}_i^{(\ell)}\|_{\ell_2} \boldsymbol{\vartheta}_i^{(\ell)}$ (see Definition 5.3.2).

The subspace detection property holds if we can prove the existence of vectors \mathbf{c} and $\boldsymbol{\nu}$ as in Lemma 7.2.1 for problems $P(\mathbf{x}_i^{(\ell)}, \mathbf{X}_{(-i)})$ of the form

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^{N-1}} \|\boldsymbol{\beta}\|_{\ell_1} \quad \text{subject to} \quad \mathbf{X}_{(-i)} \boldsymbol{\beta} = \mathbf{x}_i^{(\ell)}. \tag{7.2.1}$$

We set to prove that the vectors $\mathbf{c} = [\mathbf{0}, \dots, \mathbf{0}, \mathbf{c}_i^{(\ell)}, \mathbf{0}, \dots, \mathbf{0}]$, which is feasible for (7.3.1), and $\boldsymbol{\nu}_i^{(\ell)}$ are indeed as in Lemma 7.2.1. To do this, we have to check that the following conditions are satisfied:

$$(\mathbf{X}_{(-i)}^{(\ell)})_S^T \boldsymbol{\nu}_i^{(\ell)} = \text{sgn}(\mathbf{c}_i^{(\ell)}), \quad (7.2.2)$$

$$\left\| (\mathbf{X}_{(-i)}^{(\ell)})_{S^c}^T \boldsymbol{\nu}_i^{(\ell)} \right\|_{\ell_\infty} \leq 1, \quad (7.2.3)$$

and for $k = 1, \dots, \ell - 1, \ell + 1, \dots, L$

$$\left\| \mathbf{X}_{(-i)}^{(k)} \boldsymbol{\nu}_i^{(\ell)} \right\|_{\ell_\infty} < 1. \quad (7.2.4)$$

Conditions (7.2.2) and (7.2.3) are satisfied by definition, since

$$(\mathbf{X}_{(-i)}^{(\ell)})_S^T \boldsymbol{\nu}_i^{(\ell)} = (\mathbf{A}_{(-i)}^{(\ell)})_S^T \mathbf{U}^{(\ell)T} \mathbf{U}^{(\ell)} \boldsymbol{\vartheta}_i^{(\ell)} = (\mathbf{A}_{(-i)}^{(\ell)})_S^T \boldsymbol{\vartheta}_i^{(\ell)} = \text{sgn}(\mathbf{c}_i^{(\ell)}),$$

and

$$\left\| (\mathbf{X}_{(-i)}^{(\ell)})_{S^c}^T \boldsymbol{\nu}_i^{(\ell)} \right\|_{\ell_\infty} = \left\| (\mathbf{A}_{(-i)}^{(\ell)})_{S^c}^T \mathbf{U}^{(\ell)T} \mathbf{U}^{(\ell)} \boldsymbol{\vartheta}_i^{(\ell)} \right\|_{\ell_\infty} = \left\| (\mathbf{A}_{(-i)}^{(\ell)})_{S^c}^T \boldsymbol{\vartheta}_i^{(\ell)} \right\|_{\ell_\infty} \leq 1.$$

Therefore, in order to prove that the subspace detection property holds, it remains to check that

$$\left\| \mathbf{X}^{(k)} \boldsymbol{\nu}_i^{(\ell)} \right\|_{\ell_\infty} = \left\| \mathbf{X}^{(k)} \boldsymbol{\vartheta}_i^{(\ell)} \right\|_{\ell_\infty} \left\| \boldsymbol{\vartheta}_i^{(\ell)} \right\|_{\ell_2} < 1.$$

By definition of $\boldsymbol{\vartheta}_i^{(\ell)}$, $\left\| \mathbf{A}_{(-i)}^{(\ell)T} \boldsymbol{\vartheta}_i^{(\ell)} \right\|_{\ell_\infty} \leq 1$, and therefore, $\boldsymbol{\vartheta}_i^{(\ell)} \in (\mathcal{P}_{-i}^\ell)^o$, where

$$(\mathcal{P}_{-i}^\ell)^o = \left\{ \mathbf{z} : \left\| \mathbf{A}_{(-i)}^{(\ell)T} \mathbf{z} \right\|_{\ell_\infty} \leq 1 \right\}.$$

Definition 7.2.2 (circumradius) *The circumradius of a convex body \mathcal{P} , denoted by $R(\mathcal{P})$, is defined as the radius of the smallest ball containing \mathcal{P} .*

Using this definition and the fact that $\boldsymbol{\vartheta}_i^\ell \in (\mathcal{P}_{-i}^\ell)^o$ we have

$$\|\boldsymbol{\vartheta}_i^{(\ell)}\|_{\ell_2} \leq R(\mathcal{P}_{-i}^{\ell o}) = \frac{1}{r(\mathcal{P}_{-i}^\ell)},$$

where the equality is a consequence of the lemma below.

Lemma 7.2.3 *For a symmetric convex body \mathcal{P} , i.e. $\mathcal{P} = -\mathcal{P}$, the following relationship between the inradius of \mathcal{P} and circumradius of its polar \mathcal{P}^o holds:*

$$r(\mathcal{P})R(\mathcal{P}^o) = 1.$$

In summary, it suffices to verify that for all triples (ℓ, k, i) in which $k \neq \ell$, we have

$$\|\mathbf{X}^{(k)T} \boldsymbol{v}_i^{(\ell)}\|_{\ell_\infty} < r(\mathcal{P}_{-i}^\ell).$$

Now notice that the latter is precisely the sufficient condition given in the statement of Theorem 5.3.5, thereby concluding the proof.

7.2.2 Proof of Theorem 5.3.6

We prove this in two steps.

Step 1: We develop a lower bound about the inradii, namely,

$$\mathbb{P}\left\{\frac{c(\rho_\ell)\sqrt{\log \rho_\ell}}{\sqrt{2d_\ell}} \leq r(\mathcal{P}_{-i}^\ell) \text{ for all pairs } (\ell, i)\right\} \geq 1 - \sum_{\ell=1}^L N_\ell e^{-\sqrt{\rho_\ell} d_\ell}. \quad (7.2.5)$$

Step 2: Notice that $\mu(\mathcal{X}_\ell) = \max_{k:k \neq \ell} \|\mathbf{X}^{(k)T} \mathbf{V}^{(\ell)}\|_{\ell_\infty}$. Therefore we develop an upper bound about the subspace incoherence, namely,

$$\begin{aligned} \mathbb{P}\left\{\|\mathbf{X}^{(k)T} \mathbf{V}^{(\ell)}\|_{\ell_\infty} \leq 4(\log[N_\ell(N_k+1)] + \log L + t) \frac{\text{aff}(S_k, S_\ell)}{\sqrt{d_k \vee d_\ell}} \text{ for all pairs } (\ell, k) \text{ with } \ell \neq k\right\} \\ \geq 1 - \frac{1}{L^2} \sum_{k \neq \ell} \frac{4}{(N_k + 1)N_\ell} e^{-2t}. \end{aligned} \quad (7.2.6)$$

Notice that if the condition (5.3.2) in Theorem 5.3.6 holds, i.e.

$$\max_{k \neq \ell} 4\sqrt{2} \left(\log[N_\ell(N_k + 1)] + \log L + t \right) \text{aff}(S_k, S_\ell) < c(\rho_\ell) \sqrt{\log \rho_\ell},$$

then Step 1 and Step 2 imply that the deterministic condition in Theorem 5.3.5 holds with high probability. In turn, this gives the subspace detection property.

7.2.2.1 Proof of Step 1

Here, we simply make use of a lemma stating that the inradius of a polytope with vertices chosen uniformly at random from the unit sphere is lower bounded with high probability.

Lemma 7.2.4 ([12]) *Assume $\{P_i\}_{i=1}^N$ are independent random vectors on \mathbb{S}^{n-1} , and set $\mathcal{K} = \text{conv}(\pm P_1, \dots, \pm P_N)$. For every $\delta > 0$, there exists a constant $C(\delta)$ such that if $(1 + \delta)d < N < de^{\frac{d}{2}}$, then*

$$\mathbb{P} \left\{ r(\mathcal{K}) < \min\{C(\delta), 1/\sqrt{8}\} \sqrt{\frac{\log \frac{N}{d}}{d}} \right\} \leq e^{-d}.$$

Furthermore, there exists a numerical constant δ_0 such that for all $N > d(1 + \delta_0)$ we have

$$\mathbb{P} \left\{ r(\mathcal{K}) < \frac{1}{\sqrt{8}} \sqrt{\frac{\log \frac{N}{d}}{d}} \right\} \leq e^{-d}.$$

One can increase the probability with which this lemma holds by introducing a parameter $0 \leq \beta \leq 1$ in the lower bound. A modification of the arguments yields (note the smaller bound on the probability of failure)

$$\mathbb{P} \left\{ r(\mathcal{K}) < \min\{C(\delta), 1/\sqrt{8}\} \sqrt{\beta \frac{\log \frac{N}{d}}{d}} \right\} \leq e^{-cd^\beta N^{1-\beta}}.$$

This is where the definition of the constant $c(\rho)$ ¹ come in. We set $c(\rho) = \min\{C(\rho - 1), 1/\sqrt{8}\}$ and $\rho_0 = \delta_0 + 1$ where δ_0 is as in the above Lemma and use $\beta = \frac{1}{2}$. Now since \mathcal{P}_{-i}^ℓ consists of $2(N_\ell - 1)$ vertices on \mathbb{S}^{d_ℓ} taken from the intersection of the unit sphere with the subspace S_ℓ of dimension d_ℓ , applying Lemma 7.2.4 and using the union bound establishes (7.2.5).

7.2.2.2 Proof of Step 2

By definition

$$\|\mathbf{X}^{(k)T} \mathbf{V}^{(\ell)}\|_{\ell_\infty} = \max_{i=1,\dots,N_\ell} \|\mathbf{X}^{(k)T} \mathbf{v}_i^{(\ell)}\|_{\ell_\infty} = \max_{i=1,\dots,N_\ell} \left\| \mathbf{A}^{(k)T} \mathbf{U}^{(k)T} \mathbf{U}^{(\ell)} \frac{\vartheta_i^{(\ell)}}{\|\vartheta_i^{(\ell)}\|_{\ell_2}} \right\|_{\ell_\infty} \quad (7.2.7)$$

Now it follows from the uniform distribution of the points on each subspace that the columns of $\mathbf{A}^{(k)}$ are independently and uniformly distributed on the unit sphere of \mathbb{R}^{d_k} . Furthermore, the normalized dual points² $\vartheta_i^{(\ell)} / \|\vartheta_i^{(\ell)}\|_{\ell_2}$ are also distributed uniformly at random on the unit sphere of \mathbb{R}^{d_ℓ} . To justify this claim, assume \mathbf{U} is an orthogonal transform on \mathbb{R}^{d_ℓ} and $\vartheta_i^{(\ell)}(\mathbf{U})$ is the dual point corresponding to $\mathbf{U}\mathbf{a}_i$ and $\mathbf{U}\mathbf{A}_{(-i)}^{(\ell)}$. Then

$$\vartheta_i^{(\ell)}(\mathbf{U}) = \vartheta(\mathbf{U}\mathbf{a}_i, \mathbf{U}\mathbf{A}_{(-i)}^{(\ell)}) = \mathbf{U}\vartheta(\mathbf{a}_i, \mathbf{A}_{(-i)}^{(\ell)}) = \mathbf{U}\vartheta_i^{(\ell)}, \quad (7.2.8)$$

where we have used the fact that $\vartheta_i^{(\ell)}$ is the dual variable in the corresponding optimization problem. On the other hand we know that

$$\vartheta_i^{(\ell)}(\mathbf{U}) = \vartheta(\mathbf{U}\mathbf{a}_i, \mathbf{U}\mathbf{A}_{(-i)}^{(\ell)}) \sim \vartheta(\mathbf{a}_i, \mathbf{A}_{(-i)}^{(\ell)}), \quad (7.2.9)$$

where $X \sim Y$ means that the random variables X and Y have the same distribution. This follows from $\mathbf{U}\mathbf{a}_i \sim \mathbf{a}_i$ and $\mathbf{U}\mathbf{A}_{(-i)}^{(\ell)} \sim \mathbf{A}_{(-i)}^{(\ell)}$ since the columns of $\mathbf{A}^{(\ell)}$ are chosen

¹Recall that $c(\rho)$ is defined as a constant obeying the following two properties: (i) for all $\rho > 1$, $c(\rho) > 0$; (ii) there is a numerical value ρ_0 , such that for all $\rho \geq \rho_0$, one can take $c(\rho) = \frac{1}{\sqrt{8}}$.

²Since the columns of $\mathbf{A}^{(\ell)}$ are independently and uniformly distributed on the unit sphere of \mathbb{R}^{d_ℓ} , $\lambda_i^{(\ell)}$ in Definition 5.3.1 is uniquely defined with probability 1.

uniformly at random on the unit sphere. Combining (7.2.8) and (7.2.9) implies that for any orthogonal transformation \mathbf{U} , we have

$$\boldsymbol{\vartheta}_i^{(\ell)}(\mathbf{U}) \sim \mathbf{U} \boldsymbol{\vartheta}_i^{(\ell)},$$

which proves the claim.

Continuing with (7.2.7), since $\boldsymbol{\vartheta}_i^{(\ell)}$ and $\mathbf{A}^{(k)}$ are independent, applying Lemma 7.3.1 below with $\Delta = N_\ell L^{1.5}$, $N_1 = N_k$, $d_1 = d_k$, and $d_2 = d_\ell$ gives

$$\left\| \mathbf{A}^{(k)T} (\mathbf{U}^{(k)T} \mathbf{U}^{(\ell)}) \frac{\boldsymbol{\vartheta}_i^{(\ell)}}{\|\boldsymbol{\vartheta}_i^{(\ell)}\|_{\ell_2}} \right\|_{\ell_\infty} \leq 4(\log[N_\ell(N_k + 1)] + \log L + t) \frac{\|\mathbf{U}^{(k)T} \mathbf{U}^{(\ell)}\|_F}{\sqrt{d_k} \sqrt{d_\ell}},$$

with probability at least $1 - \frac{2}{(N_k + 1)N_\ell^2 L^2} e^{-2t}$. Finally, applying the union bound twice gives (7.2.6).

Lemma 7.2.5 *Let $\mathbf{A} \in \mathbb{R}^{d_1 \times N_1}$ be a matrix with columns sampled uniformly at random from the unit sphere of \mathbb{R}^{d_1} , $\boldsymbol{\vartheta} \in \mathbb{R}^{d_2}$ be a vector sampled uniformly at random from the unit sphere of \mathbb{R}^{d_2} and independent of \mathbf{A} and $\Sigma \in \mathbb{R}^{d_1 \times d_2}$ be a deterministic matrix obeying $\|\Sigma\| \leq 1$. We have*

$$\|\mathbf{A}^T \Sigma \boldsymbol{\vartheta}\|_{\ell_\infty} \leq 4(\log(N_1 + 1) + \log \Delta + t) \frac{\|\Sigma\|_F}{\sqrt{d_1} \sqrt{d_2}},$$

with probability at least $1 - \frac{4}{(N_1 + 1)\Delta^2} e^{-2t}$.

Proof The proof is standard. Without loss of generality, we assume $d_1 \leq d_2$ as the other case is similar. To begin with, the mapping $\lambda \mapsto \|\Sigma \boldsymbol{\lambda}\|_{\ell_2}$ is Lipschitz with constant at most σ_1 (this is the largest singular value of Σ). Hence, Borell's inequality gives

$$\mathbb{P}\left\{ \|\Sigma \boldsymbol{\lambda}\|_{\ell_2} - \sqrt{\mathbb{E}\|\Sigma \boldsymbol{\lambda}\|_{\ell_2}^2} \geq \epsilon \right\} < e^{-d_2 \epsilon^2 / (2\sigma_1^2)}.$$

Because $\boldsymbol{\lambda}$ is uniformly distributed on the unit sphere, we have $\mathbb{E}\|\Sigma \boldsymbol{\lambda}\|_{\ell_2}^2 = \|\Sigma\|_F^2 / d_2$. Plugging $\epsilon = (b-1) \frac{\|\Sigma\|_F}{\sqrt{d_2}}$ into the above inequality, where $b = 2\sqrt{\log(N_1 + 1) + \log \Delta + t}$,

and using $\|\Sigma\|_F/\sigma_1 \geq 1$ give

$$\mathbb{P}(\|\Sigma\lambda\|_{\ell_2} > b \frac{\|\Sigma\|_F}{\sqrt{d_2}}) \leq \frac{2}{(N_1 + 1)^2 \Delta^2} e^{-2t}.$$

Further, letting $\mathbf{a} \in \mathbb{R}^{d_1}$ be a representative column of \mathbf{A} , a well-known upper bound on the area of spherical caps gives

$$\mathbb{P}\left\{ |\mathbf{a}^T \mathbf{z}| > \epsilon \|\mathbf{z}\|_{\ell_2} \right\} \leq 2e^{\frac{-d_1 \epsilon^2}{2}}$$

in which \mathbf{z} is a fixed vector. We use $\mathbf{z} = \Sigma\lambda$, and $\epsilon = b/\sqrt{d_1}$. Therefore, for any column \mathbf{a} of \mathbf{A} we have

$$\mathbb{P}\left\{ |\mathbf{a}^T \Sigma\lambda| > \frac{b}{\sqrt{d_1}} \|\Sigma\lambda\|_{\ell_2} \right\} \leq 2e^{\frac{-d_1 \epsilon^2}{2}} = \frac{2}{(N_1 + 1)^2 \Delta^2} e^{-2t}.$$

Now applying the union bound yields

$$\mathbb{P}\left(\|\mathbf{A}^T \Sigma\lambda\|_{\ell_\infty} > \frac{b}{\sqrt{d_1}} \|\Sigma\lambda\|_{\ell_2} \right) \leq \frac{2}{(N_1 + 1) \Delta^2} e^{-2t}.$$

Plugging in the bound for $\|\Sigma\lambda\|_{\ell_2}$ concludes the proof. ■

7.2.3 Proof of Theorem 5.3.7

We prove this in two steps.

Step 1: We use the lower bound about the inradii used in Step 1 of the proof of Theorem 5.3.6 with $\beta = \frac{1}{2}$, namely,

$$\mathbb{P}\left\{ \frac{c(\rho)}{\sqrt{2}} \sqrt{\frac{\log \rho}{d}} \leq r(\mathcal{P}_i^\ell) \text{ for all pairs } (\ell, i) \right\} \geq 1 - Ne^{-\sqrt{\rho}d}.$$

Step 2: We develop an upper bound about subspace incoherence, namely,

$$\mathbb{P}\left\{ \mu(\mathcal{X}_\ell) \leq \sqrt{\frac{6 \log N}{n}} \text{ for all } \ell \right\} \geq 1 - \frac{2}{N}.$$

To prove Step 2, notice that in the fully random model, the marginal distribution of a column \mathbf{x} is uniform on the unit sphere. Furthermore, since the points on each subspace are sampled uniformly at random, the argument in the proof of Theorem 5.3.6 asserts that the dual directions are sampled uniformly at random on each subspace. By what we have just seen, the points $\mathbf{v}_i^{(\ell)}$ are then also distributed uniformly at random on the unit sphere (they are not independent). Lastly, the random vectors $\mathbf{v}_i^{(\ell)}$ and $\mathbf{x} \in \mathcal{X} \setminus \mathcal{X}_\ell$ are independent. The distribution of their inner product is as if one were fixed, and applying the well-known upper bound on the area of a spherical cap gives

$$\mathcal{P}\left\{\left|\langle \mathbf{x}, \mathbf{v}_i^{(\ell)} \rangle\right| \geq \sqrt{\frac{6 \log N}{n}}\right\} \leq \frac{2}{N^3}.$$

Step 2 follows by applying the union bound to at most N^2 such pairs.

7.3 Proof of results with noise

We prove our results on the effectiveness of RSC-N in this section. Throughout we use L_m to denote $\log m$ up to a fixed numerical constant. The value of this constant may change from line to line. Likewise, C is a generic numerical constant whose value may change at each occurrence.

Next, we work with $\mathbf{y} := \mathbf{y}_1$ for convenience, assumed to originate from S_1 , which is no loss of generality. It is also convenient to partition \mathbf{Y} as $\{\mathbf{Y}^{(1)}, \mathbf{Y}^{(2)}, \dots, \mathbf{Y}^{(L)}\}$, where for each ℓ , $\mathbf{Y}^{(\ell)}$ are those noisy columns from subspace S_ℓ ; when $\ell = 1$, we exclude the response \mathbf{y}_1 from $\mathbf{Y}^{(1)}$. With this notation, the problem (4.4.2) takes the form

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^{N-1}} \quad \frac{1}{2} \|\mathbf{y} - (\mathbf{Y}^{(1)}\boldsymbol{\beta}^{(1)} + \dots + \mathbf{Y}^{(L)}\boldsymbol{\beta}^{(L)})\|_{\ell_2}^2 + \lambda \|\boldsymbol{\beta}^{(1)}\|_{\ell_1} + \dots + \lambda \|\boldsymbol{\beta}^{(L)}\|_{\ell_1}. \quad (7.3.1)$$

Throughout, \mathbf{y}_{\parallel} denotes the projection of the vector \mathbf{y} onto the subspace S_1 . Similarly, $\mathbf{Y}_{\parallel}^{(1)}$ is the projection of the columns of the matrix $\mathbf{Y}^{(1)}$ onto S_1 . Similarly, we use \mathbf{y}_{\perp} to denote projection onto the orthogonal complement S_1^\perp ; hence, $\mathbf{y} = \mathbf{y}_{\parallel} + \mathbf{y}_{\perp}$ and

$\mathbf{Y}^{(1)} = \mathbf{Y}_{\parallel}^{(1)} + \mathbf{Y}_{\perp}^{(1)}$. Moreover, $\mathbf{U}_1 \in \mathbb{R}^{n \times d}$ and $\mathbf{U}_1^\perp \in \mathbb{R}^{n \times (n-d)}$ are orthonormal bases for S_1 and S_1^\perp .

Since $\mathbf{y}_{\parallel} = \mathbf{x} + \mathbf{z}_{\parallel}$ with $\|\mathbf{x}\|_{\ell_2} = 1$ and $\mathbb{E} \|\mathbf{z}_{\parallel}\|_{\ell_2}^2 = \sigma^2 d/n$, it is obvious that under the stated assumptions, $\|\mathbf{y}_{\parallel}\|_{\ell_2} \in [3/4, 5/4]$ with very high probability as shown in Lemma B.0.15. The same applies to all the columns of $\mathbf{Y}_{\parallel}^{(1)}$. From now on, we will operate under these two assumptions, which hold simultaneously over an event of probability at least $1 - 1/N^2$.

The sampling condition shall be required many times in the proofs. Specifically, it is used to establish the correct scaling of the inradius (Lemma C.0.18), which in turn is used to establish the correct scaling of the ℓ_1 norm of the primal solution and the ℓ_2 norm of the dual solution (Lemmas C.0.17 and C.0.19). Both these estimates play a crucial role in numerous arguments. To avoid repetition, we refrain from stating the lower bound on ρ each time.

7.3.1 Intermediate results

In this section, we record a few important results that shall be used to establish the no-false and many true discoveries theorems. Now the reader interested in our proofs may first want to pass over this section rather quickly, and return to it once it is clear how our arguments reduce to the technical lemmas below.

7.3.1.1 Preliminaries

Our first lemma rephrases Lemma 7.3.1 and bounds the size of the dot product between random vectors. We omit the proof.

Lemma 7.3.1 *Let $\mathbf{A} \in \mathbb{R}^{d_1 \times N_1}$ be a matrix with columns sampled uniformly at random from the unit sphere of \mathbb{R}^{d_1} , $\mathbf{w} \in \mathbb{R}^{d_2}$ be a vector sampled uniformly at random from the unit sphere of \mathbb{R}^{d_2} and independent of \mathbf{A} and $\Sigma \in \mathbb{R}^{d_1 \times d_2}$ be a deterministic matrix. We have*

$$\|\mathbf{A}^T \Sigma \mathbf{w}\|_{\ell_\infty} \leq \sqrt{\log a \log b} \frac{\|\Sigma\|_F}{\sqrt{d_1} \sqrt{d_2}}, \quad (7.3.2)$$

with probability at least $1 - \frac{2}{\sqrt{a}} - \frac{2N_1}{\sqrt{b}}$.

We are interested in this because (7.3.2) relates the size of the dot products with the affinity between subspaces as follows: suppose the unit-norm vector \mathbf{x}_i is drawn uniformly at random from S_i , then

$$\mathbf{X}^{(j)T} \mathbf{x}_i = \mathbf{A}^T \Sigma \mathbf{w};$$

\mathbf{A} and \mathbf{w} are as in the lemma and $\Sigma = \mathbf{U}^{(j)T} \mathbf{U}^{(i)}$, where $\mathbf{U}^{(j)}$ (resp. $\mathbf{U}^{(i)}$) is an orthobasis for S_j (resp. S_i). By definition, $\|\Sigma\|_F = \sqrt{d_j \wedge d_i} \text{aff}(S_j, S_i)$.

7.3.1.2 The first step of Algorithm 5

As claimed in Section 4.4.1, the first step of Algorithm 5 returns an optimal value that is a reasonable proxy for the unknown dimension.

Lemma 7.3.2 *Let $\text{Val}(\text{Step 1})$ be the optimal value of (4.4.4) with $\tau = 2\sigma$. Assume $\rho_1 > \rho^*$ as earlier. Then*

$$\frac{1}{10} \sqrt{\frac{d_1}{\log \rho_1}} \leq \text{Val}(\text{Step 1}) \leq 2\sqrt{d_1}. \quad (7.3.3)$$

The upper bound holds with probability at least $1 - e^{\gamma_1 n} - e^{-\gamma_2 d_1}$. The lower bound holds with probability at least $1 - e^{-\gamma_3 d_1} - \frac{10}{N^2}$.

Proof We begin with the upper bound. Let $\boldsymbol{\beta}_0 = \mathbf{X}^{(1)T} (\mathbf{X}^{(1)} \mathbf{X}^{(1)T})^{-1} \mathbf{x}$ be the minimum ℓ_2 -norm solution to the noiseless problem $\mathbf{X}\boldsymbol{\beta}_0 = \mathbf{x}$. We show that $\boldsymbol{\beta}_0$ is feasible for (4.4.4). We have

$$\mathbf{y} - \mathbf{Y}\boldsymbol{\beta}_0 = \mathbf{x} - \mathbf{X}\boldsymbol{\beta}_0 + (\mathbf{z} - \mathbf{Z}\boldsymbol{\beta}_0) = \mathbf{z} - \mathbf{Z}\boldsymbol{\beta}_0,$$

which gives

$$\mathcal{L}(\mathbf{z} - \mathbf{Z}\boldsymbol{\beta}_0 | \mathbf{X}, \mathbf{x}) = \mathcal{N}(0, V \mathbf{I}_n), \quad V = (1 + \|\boldsymbol{\beta}_0\|_{\ell_2}^2) \sigma^2 / n$$

(the notation $\mathcal{L}(Y|X)$ is the conditional law of Y given X). Hence, the conditional distribution of $\|\mathbf{z} - \mathbf{Z}\beta_0\|_{\ell_2}^2$ is that of a chi square and (B.0.1) gives

$$\|\mathbf{z} - \mathbf{Z}\beta_0\|_{\ell_2} \leq \sqrt{2(1 + \|\beta_0\|_{\ell_2}^2)}\sigma$$

with probability at least $1 - e^{-\gamma_1 n}$. On the other hand,

$$\|\beta_0\|_{\ell_2} \leq \frac{\|\mathbf{x}\|_{\ell_2}}{\sigma_{\min}(\mathbf{X}^{(1)})}$$

and applying Lemma B.0.13 gives

$$\|\beta_0\|_{\ell_2} \leq \frac{1}{\sqrt{\frac{N_1}{d_1}} - 2},$$

which holds with probability at least $1 - e^{-\gamma_2 d_1}$. If $N_1 > 9d_1$, then $\|\beta_0\|_{\ell_2} \leq 1$ and thus β_0 is feasible. Therefore,

$$\|\beta^*\|_{\ell_1} \leq \|\beta_0\|_{\ell_1} \leq \sqrt{N_1} \|\beta_0\|_{\ell_2} \leq \frac{\sqrt{d_1}}{1 - 2\sqrt{\frac{d_1}{N_1}}} \leq 2\sqrt{d_1},$$

where the last inequality holds provided that $N_1 \geq 16d_1$.

We now turn to the lower bound and let β^* be an optimal solution. Notice that $\|\mathbf{y}_\parallel - \mathbf{Y}_\parallel \beta^*\|_{\ell_2} \leq \|\mathbf{y} - \mathbf{Y} \beta^*\|_{\ell_2} \leq 2\sigma$ so that β^* is feasible for

$$\min \|\beta\|_{\ell_1} \quad \text{subject to} \quad \|\mathbf{y}_\parallel - \mathbf{Y}_\parallel \beta\|_{\ell_2} \leq 2\sigma. \quad (7.3.4)$$

We bound the optimal value of this program from below. The dual of (7.3.4) is

$$\max \langle \mathbf{y}_\parallel, \boldsymbol{\nu} \rangle - 2\sigma \|\boldsymbol{\nu}\|_{\ell_2} \quad \text{subject to} \quad \|\mathbf{Y}_\parallel^T \boldsymbol{\nu}\|_{\ell_\infty} \leq 1. \quad (7.3.5)$$

Slater's condition holds and the primal and dual optimal values are equal. To simplify notation set $\mathbf{A} = \mathbf{Y}_\parallel^{(1)}$. Define

$$\boldsymbol{\nu}^* \in \arg \max \langle \mathbf{y}_\parallel, \boldsymbol{\nu} \rangle \quad \text{subject to} \quad \|\mathbf{A}^T \boldsymbol{\nu}\|_{\ell_\infty} \leq 1.$$

Notice that $\boldsymbol{\nu}^*$ has random direction on subspace S_1 . Therefore, combining Lemmas 7.3.1, B.0.12, and C.0.19 together with the affinity condition implies that for $\ell \neq 1$, $\|\mathbf{Y}_{\parallel}^{(\ell)^T} \boldsymbol{\nu}^*\|_{\ell_\infty} \leq 1$ with high probability. In short, $\boldsymbol{\nu}^*$ is feasible for (7.3.5).

Since \mathbf{y}_{\parallel} has random direction, the arguments (with $t = 1/6$) in Step 2 of the proof of Theorem 5.5.2 (Section 7.4.2.1.2) shall give

$$\langle \mathbf{y}_{\parallel}, \boldsymbol{\nu}^* \rangle \geq \frac{1}{\sqrt{2\pi e}} \sqrt{\frac{d_1}{\log \rho_1}}.$$

Also, by Lemma C.0.19,

$$\|\boldsymbol{\nu}^*\|_{\ell_2} \leq \frac{16}{3} \sqrt{\frac{d_1}{\log \rho_1}}.$$

Since $\boldsymbol{\nu}^*$ is feasible for (7.3.5), the optimal value of (7.3.5) is greater or equal than

$$\langle \mathbf{y}_{\parallel}, \boldsymbol{\nu}^* \rangle - 2\sigma \|\boldsymbol{\nu}^*\|_{\ell_2} \geq \frac{1}{10} \sqrt{\frac{d_1}{\log \rho_1}},$$

where the inequality follows from the bound on the noise level. This concludes the proof. ■

7.3.1.3 The reduced and projected problems

When there are no false discoveries, the solution to (7.3.1) coincides with that of the *reduced problem*

$$\hat{\boldsymbol{\beta}}^{(1)} \in \arg \min \frac{1}{2} \|\mathbf{y} - \mathbf{Y}^{(1)} \boldsymbol{\beta}^{(1)}\|_{\ell_2}^2 + \lambda \|\boldsymbol{\beta}^{(1)}\|_{\ell_1}. \quad (7.3.6)$$

Not surprisingly, we need to analyze the properties of the solution to this problem. In particular, we would like to understand something about the orientation and the size of the residual vector $\mathbf{y} - \mathbf{Y}^{(1)} \hat{\boldsymbol{\beta}}^{(1)}$.

A problem close to (7.3.6) is the *projected problem*

$$\tilde{\boldsymbol{\beta}}^{(1)} \in \arg \min \frac{1}{2} \|\mathbf{y}_{\parallel} - \mathbf{Y}_{\parallel}^{(1)} \boldsymbol{\beta}^{(1)}\|_{\ell_2}^2 + \lambda \|\boldsymbol{\beta}^{(1)}\|_{\ell_1}. \quad (7.3.7)$$

The difference with the reduced problem is that the goodness of fit only involves the residual sum of squares of the projected residuals. Intuitively, the solutions to the two problems (7.3.6) and (7.3.7) should be close. Our strategy is to gain some insights about the solution to the reduced problem by studying the properties of the projected problem.

7.3.1.4 Properties of the projected problem

The sole purpose of this subsection is to state this:

Lemma 7.3.3 *Let $\tilde{\beta}^{(1)}$ be any solution to the projected problem and assume that $N_1/d_1 \geq \rho^*$ as before. Then there exists an absolute constant C such that for all $\lambda > 0$,*

$$\|\tilde{\beta}^{(1)}\|_{\ell_2} \leq C \quad (7.3.8)$$

holds with probability at least $1 - 5e^{-\gamma_1 d_1} - e^{-\sqrt{N_1 d_1}}$.

This estimate shall play a crucial role in our arguments. It is a consequence of sharp estimates obtained by Wojtaszczyk [244] in the area of compressed sensing. As not to interrupt the flow, we postpone its proof to Section 7.3.3.

In the asymptotic regime ($\rho_1 = N_1/d_1$ fixed and $d_1 \rightarrow \infty$), one can sharpen the upper bound (7.3.8) by taking $C = 1$. This leverages the asymptotic theory developed in [31] and [33] as explained in Appendix D.

7.3.1.5 Properties of the reduced problem

We now collect two important facts about the residuals to the reduced problem. The first concerns their orientation.

Lemma 7.3.4 (Isotropy of the residuals) *The projection of the residual vector $r = y - Y^{(1)}\hat{\beta}^{(1)}$ onto either S_1 or S_1^\perp has uniform orientation.*

Proof Consider any unitary transformation \mathbf{U}^\parallel (resp. \mathbf{U}^\perp) leaving S_1 (resp. S_1^\perp) invariant. Since

$$\begin{aligned} \frac{1}{2} \left\| \mathbf{U}^\parallel (\mathbf{y}_\parallel - \mathbf{Y}_\parallel^{(1)} \hat{\boldsymbol{\beta}}^{(1)}) \right\|_{\ell_2}^2 + \frac{1}{2} \left\| \mathbf{U}^\perp (\mathbf{y}_\perp - \mathbf{Y}_\perp^{(1)} \hat{\boldsymbol{\beta}}^{(1)}) \right\|_{\ell_2}^2 + \lambda \|\hat{\boldsymbol{\beta}}^{(1)}\|_{\ell_1} \\ = \frac{1}{2} \left\| \mathbf{y}_\parallel - \mathbf{Y}_\parallel^{(1)} \hat{\boldsymbol{\beta}}^{(1)} \right\|_{\ell_2}^2 + \frac{1}{2} \left\| \mathbf{y}_\perp - \mathbf{Y}_\perp^{(1)} \hat{\boldsymbol{\beta}}^{(1)} \right\|_{\ell_2}^2 + \lambda \|\hat{\boldsymbol{\beta}}^{(1)}\|_{\ell_1}, \end{aligned}$$

the LASSO functional is invariant and this gives

$$\hat{\boldsymbol{\beta}}^{(1)}(\mathbf{U}^\parallel \mathbf{y}_\parallel, \mathbf{U}^\perp \mathbf{y}_\perp, \mathbf{U}^\parallel \mathbf{Y}_\parallel^{(1)}, \mathbf{U}^\perp \mathbf{Y}_\perp^{(1)}) = \hat{\boldsymbol{\beta}}^{(1)}(\mathbf{y}_\parallel, \mathbf{y}_\perp, \mathbf{Y}_\parallel^{(1)}, \mathbf{Y}_\perp^{(1)}).$$

Let $\mathbf{r}^\parallel(\mathbf{y}_\parallel, \mathbf{Y}_\parallel^{(1)}) = \mathbf{y}_\parallel - \mathbf{Y}_\parallel^{(1)} \hat{\boldsymbol{\beta}}^{(1)}$ and $\mathbf{r}^\perp(\mathbf{y}_\perp, \mathbf{Y}_\perp^{(1)}) = \mathbf{y}_\perp - \mathbf{Y}_\perp^{(1)} \hat{\boldsymbol{\beta}}^{(1)}$ be the projections of the residuals. Since \mathbf{y}_\parallel and $\mathbf{Y}_\parallel^{(1)}$ are invariant under rotations leaving S_1 invariant, we have

$$\begin{aligned} \mathbf{r}^\parallel(\mathbf{U}^\parallel \mathbf{y}_\parallel, \mathbf{U}^\parallel \mathbf{Y}_\parallel^{(1)}) &= \mathbf{U}^\parallel \mathbf{y}_\parallel - \mathbf{U}^\parallel \mathbf{Y}_\parallel^{(1)} \hat{\boldsymbol{\beta}}^{(1)}(\mathbf{U}^\parallel \mathbf{y}_\parallel, \mathbf{U}^\perp \mathbf{y}_\perp, \mathbf{U}^\parallel \mathbf{Y}_\parallel^{(1)}, \mathbf{U}^\perp \mathbf{Y}_\perp^{(1)}) \\ &= \mathbf{U}^\parallel \mathbf{y}_\parallel - \mathbf{U}^\parallel \mathbf{Y}_\parallel^{(1)} \hat{\boldsymbol{\beta}}^{(1)}(\mathbf{y}_\parallel, \mathbf{y}_\perp, \mathbf{Y}_\parallel^{(1)}, \mathbf{Y}_\perp^{(1)}) \\ &\sim \mathbf{y}_\parallel - \mathbf{Y}_\parallel^{(1)} \hat{\boldsymbol{\beta}}^{(1)}(\mathbf{y}_\parallel, \mathbf{y}_\perp, \mathbf{Y}_\parallel^{(1)}, \mathbf{Y}_\perp^{(1)}) \\ &= \mathbf{r}^\parallel(\mathbf{y}_\parallel, \mathbf{Y}_\parallel^{(1)}), \end{aligned}$$

where $X \sim Y$ means that the random variables X and Y have the same distribution. Therefore, the distribution of $\mathbf{r}^\parallel(\mathbf{y}_\parallel, \mathbf{Y}_\parallel^{(1)}) = \mathbf{y}_\parallel - \mathbf{Y}_\parallel^{(1)} \hat{\boldsymbol{\beta}}^{(1)}$ is invariant under rotations leaving S_1 invariant. In other words, the projection \mathbf{r}^\parallel has uniform orientation. In a similar manner we conclude that \mathbf{r}^\perp has uniform orientation as well. \blacksquare

The next result controls the size of the residuals.

Lemma 7.3.5 (Size of residuals) *If $N_1/d_1 \geq \rho^*$, then for all $\lambda > 0$,*

$$\left\| \mathbf{y}_\perp - \mathbf{Y}_\perp^{(1)} \hat{\boldsymbol{\beta}}^{(1)} \right\|_{\ell_2} \leq C \sigma. \quad (7.3.9)$$

Also,

$$\left\| \mathbf{y}_\parallel - \mathbf{Y}_\parallel^{(1)} \hat{\boldsymbol{\beta}}^{(1)} \right\|_{\ell_2} \leq \frac{32}{3} \lambda \sqrt{\frac{d_1}{\log(N_1/d_1)}} + C\sigma. \quad (7.3.10)$$

Both these inequalities hold with probability at least $1 - e^{-\gamma_1(n-d_1)} - 5e^{-\gamma_2 d_1} - e^{-\sqrt{N_1 d_1}}$, where γ_1 and γ_2 are fixed numerical constants. Thus if $\lambda > \sigma/\sqrt{8d_1}$, then

$$\|\mathbf{y}_\parallel - \mathbf{Y}_\parallel^{(1)} \hat{\boldsymbol{\beta}}^{(1)}\|_{\ell_2} \leq C\lambda\sqrt{d_1}. \quad (7.3.11)$$

Proof We begin with (7.3.9). Since $\hat{\boldsymbol{\beta}}^{(1)}$ is optimal for the reduced problem,

$$\frac{1}{2} \|\mathbf{y} - \mathbf{Y}^{(1)} \hat{\boldsymbol{\beta}}^{(1)}\|_{\ell_2}^2 + \lambda \|\hat{\boldsymbol{\beta}}^{(1)}\|_{\ell_1} \leq \frac{1}{2} \|\mathbf{y} - \mathbf{Y}^{(1)} \tilde{\boldsymbol{\beta}}^{(1)}\|_{\ell_2}^2 + \lambda \|\tilde{\boldsymbol{\beta}}^{(1)}\|_{\ell_1}.$$

Conversely, since $\tilde{\boldsymbol{\beta}}^{(1)}$ is optimal for the projected problem,

$$\frac{1}{2} \|\mathbf{y}_\parallel - \mathbf{Y}_\parallel^{(1)} \tilde{\boldsymbol{\beta}}^{(1)}\|_{\ell_2}^2 + \lambda \|\tilde{\boldsymbol{\beta}}^{(1)}\|_{\ell_1} \leq \frac{1}{2} \|\mathbf{y}_\parallel - \mathbf{Y}_\parallel^{(1)} \hat{\boldsymbol{\beta}}^{(1)}\|_{\ell_2}^2 + \lambda \|\hat{\boldsymbol{\beta}}^{(1)}\|_{\ell_1}.$$

Now Parseval equality

$$\|\mathbf{y} - \mathbf{Y}^{(1)} \hat{\boldsymbol{\beta}}^{(1)}\|_{\ell_2}^2 = \|\mathbf{y}_\parallel - \mathbf{Y}_\parallel^{(1)} \hat{\boldsymbol{\beta}}^{(1)}\|_{\ell_2}^2 + \|\mathbf{y}_\perp - \mathbf{Y}_\perp^{(1)} \hat{\boldsymbol{\beta}}^{(1)}\|_{\ell_2}^2$$

(and similarly for $\tilde{\boldsymbol{\beta}}^{(1)}$) together with the last two inequalities give

$$\|\mathbf{y}_\perp - \mathbf{Y}_\perp^{(1)} \hat{\boldsymbol{\beta}}^{(1)}\|_{\ell_2} \leq \|\mathbf{y}_\perp - \mathbf{Y}_\perp^{(1)} \tilde{\boldsymbol{\beta}}^{(1)}\|_{\ell_2}.$$

Now observe that \mathbf{y}_\parallel , \mathbf{y}_\perp , $\mathbf{Y}_\parallel^{(1)}$ and $\mathbf{Y}_\perp^{(1)}$ are all independent from each other. Since $\tilde{\boldsymbol{\beta}}^{(1)}$ is a function of \mathbf{y}_\parallel and $\mathbf{Y}_\parallel^{(1)}$, it is independent from \mathbf{y}_\perp and $\mathbf{Y}_\perp^{(1)}$. Hence,

$$\mathcal{L}(\mathbf{U}_1^{\perp T} (\mathbf{y}_\perp - \mathbf{Y}_\perp^{(1)} \tilde{\boldsymbol{\beta}}^{(1)}) | \mathbf{y}_\parallel, \mathbf{Y}_\parallel^{(1)}) = \mathcal{N}(0, V \mathbf{I}_{n-d_1}), \quad V = \frac{\sigma^2}{n} \left(1 + \|\tilde{\boldsymbol{\beta}}^{(1)}\|_{\ell_2}^2 \right).$$

Conditionally then, it follows from the chi-square tail bound (B.0.1) with $\epsilon = 1$ that

$$\|\mathbf{y}_\perp - \mathbf{Y}_\perp^{(1)} \tilde{\boldsymbol{\beta}}^{(1)}\|_{\ell_2}^2 \leq 2\sigma^2 \left(1 - \frac{d_1}{n} \right) \left(1 + \|\tilde{\boldsymbol{\beta}}^{(1)}\|_{\ell_2}^2 \right),$$

which holds with probability at least $1 - e^{-\gamma_1(n-d_1)}$, where $\gamma_1 = \frac{(1-\log 2)}{2}$. Unconditionally, our first claim follows from Lemma 7.3.3.

We now turn our attention to (7.3.10). Our argument uses the solution to the *noiseless* projected problem

$$\bar{\beta}^{(1)} \in \arg \min \|\beta^{(1)}\|_{\ell_1} \quad \text{subject to} \quad \mathbf{y}_{\parallel} = \mathbf{Y}_{\parallel}^{(1)} \beta^{(1)}; \quad (7.3.12)$$

this is the solution to the projected problem as $\lambda \rightarrow 0^+$. With this, we proceed until (7.3.13) as in [60, 61]. Since $\hat{\beta}^{(1)}$ is optimal for the reduced problem,

$$\frac{1}{2} \|\mathbf{y} - \mathbf{Y}^{(1)} \hat{\beta}^{(1)}\|_{\ell_2}^2 + \lambda \|\hat{\beta}^{(1)}\|_{\ell_1} \leq \frac{1}{2} \|\mathbf{y} - \mathbf{Y}^{(1)} \bar{\beta}^{(1)}\|_{\ell_2}^2 + \lambda \|\bar{\beta}^{(1)}\|_{\ell_1}.$$

Put $\mathbf{h} = \hat{\beta}^{(1)} - \bar{\beta}^{(1)}$. Standard simplifications give

$$\frac{1}{2} \|\mathbf{Y}^{(1)} \mathbf{h}\|_{\ell_2}^2 + \lambda \|\bar{\beta}^{(1)} + \mathbf{h}\|_{\ell_1} \leq \langle \mathbf{y} - \mathbf{Y}^{(1)} \bar{\beta}^{(1)}, \mathbf{Y}^{(1)} \mathbf{h} \rangle + \lambda \|\bar{\beta}^{(1)}\|_{\ell_1}.$$

Letting S be the support of $\bar{\beta}^{(1)}$, we have

$$\|\bar{\beta}^{(1)} + \mathbf{h}\|_{\ell_1} = \|\bar{\beta}_S + \mathbf{h}_S\|_{\ell_1} + \|\mathbf{h}_{S^c}\|_{\ell_1} \geq \|\bar{\beta}^{(1)}\|_{\ell_1} + \langle \text{sgn}(\bar{\beta}_S), \mathbf{h}_S \rangle + \|\mathbf{h}_{S^c}\|_{\ell_1}.$$

This yields

$$\frac{1}{2} \|\mathbf{Y}^{(1)} \mathbf{h}\|_{\ell_2}^2 + \lambda \|\mathbf{h}_{S^c}\|_{\ell_1} \leq \langle \mathbf{y} - \mathbf{Y}^{(1)} \bar{\beta}^{(1)}, \mathbf{Y}^{(1)} \mathbf{h} \rangle - \lambda \langle \text{sgn}(\bar{\beta}_S^{(1)}), \mathbf{h}_S \rangle.$$

By definition, $\mathbf{y}_{\parallel} - \mathbf{Y}_{\parallel}^{(1)} \bar{\beta}^{(1)} = 0$, and thus

$$\frac{1}{2} \|\mathbf{Y}^{(1)} \mathbf{h}\|_{\ell_2}^2 + \lambda \|\mathbf{h}_{S^c}\|_{\ell_1} \leq \langle \mathbf{y}_{\perp} - \mathbf{Y}_{\perp}^{(1)} \bar{\beta}^{(1)}, \mathbf{Y}_{\perp}^{(1)} \mathbf{h} \rangle - \lambda \langle \text{sgn}(\bar{\beta}_S^{(1)}), \mathbf{h}_S \rangle. \quad (7.3.13)$$

Continue with

$$\langle \mathbf{Y}_{\perp}^{(1)} \mathbf{h}, \mathbf{y}_{\perp} - \mathbf{Y}_{\perp}^{(1)} \bar{\beta}^{(1)} \rangle \leq \|\mathbf{Y}_{\perp}^{(1)} \mathbf{h}\|_{\ell_2} \|\mathbf{y}_{\perp} - \mathbf{Y}_{\perp}^{(1)} \bar{\beta}^{(1)}\|_{\ell_2} \leq \frac{1}{2} \|\mathbf{Y}_{\perp}^{(1)} \mathbf{h}\|_{\ell_2}^2 + \frac{1}{2} \|\mathbf{y}_{\perp} - \mathbf{Y}_{\perp}^{(1)} \bar{\beta}^{(1)}\|_{\ell_2}^2$$

so that

$$\frac{1}{2} \|\mathbf{Y}_{\parallel}^{(1)} \mathbf{h}\|_{\ell_2}^2 + \lambda \|\mathbf{h}_{S^c}\|_{\ell_1} \leq \frac{1}{2} \|\mathbf{y}_{\perp} - \mathbf{Y}_{\perp}^{(1)} \bar{\beta}^{(1)}\|_{\ell_2}^2 - \lambda \langle \text{sgn}(\bar{\beta}_S^{(1)}), \mathbf{h}_S \rangle. \quad (7.3.14)$$

Now set $\mathbf{A} = \mathbf{Y}_{\parallel}^{(1)}$ for notational convenience. Since $\bar{\boldsymbol{\beta}}^{(1)}$ is optimal, there exists $\boldsymbol{\nu}$ such that

$$\mathbf{v} = \mathbf{A}^T \boldsymbol{\nu}, \quad \mathbf{v}_S = \text{sgn}(\bar{\boldsymbol{\beta}}_S^{(1)}) \text{ and } \|\mathbf{v}_{S^c}\|_{\ell_\infty} \leq 1.$$

Also, Corollary C.0.19 gives

$$\|\boldsymbol{\nu}\|_{\ell_2}^2 \leq \frac{256}{9} \frac{d_1}{\log(N_1/d_1)}. \quad (7.3.15)$$

With this

$$\langle \text{sgn}(\bar{\boldsymbol{\beta}}_S^{(1)}), \mathbf{h}_S \rangle = \langle \mathbf{v}_S, \mathbf{h}_S \rangle = \langle \boldsymbol{\nu}, \mathbf{A}\mathbf{h} \rangle - \langle \mathbf{v}_{S^c}, \mathbf{h}_{S^c} \rangle.$$

We have

$$|\langle \boldsymbol{\nu}, \mathbf{A}\mathbf{h} \rangle| \leq \|\mathbf{A}\mathbf{h}\|_{\ell_2} \|\boldsymbol{\nu}\|_{\ell_2} \leq \frac{1}{4\lambda} \|\mathbf{A}\mathbf{h}\|_{\ell_2}^2 + \lambda \|\boldsymbol{\nu}\|_{\ell_2}^2$$

and, therefore,

$$\lambda |\langle \text{sgn}(\bar{\boldsymbol{\beta}}_S^{(1)}), \mathbf{h}_S \rangle| \leq \frac{1}{4} \|\mathbf{A}\mathbf{h}\|_{\ell_2}^2 + \lambda^2 \|\boldsymbol{\nu}\|_{\ell_2}^2 + \lambda \|\mathbf{h}_{S^c}\|_{\ell_1}.$$

Plugging this into (7.3.14), we obtain

$$\frac{1}{4} \|\mathbf{A}\mathbf{h}\|_{\ell_2}^2 \leq \frac{1}{2} \left\| \mathbf{y}_\perp - \mathbf{Y}_\perp^{(1)} \bar{\boldsymbol{\beta}}^{(1)} \right\|_{\ell_2}^2 + \lambda^2 \|\boldsymbol{\nu}\|_{\ell_2}^2. \quad (7.3.16)$$

This concludes the proof since by definition $\mathbf{A}\mathbf{h} = \mathbf{y}_\parallel - \mathbf{Y}_\parallel^{(1)} \hat{\boldsymbol{\beta}}^{(1)}$ and since we already know that $\|\mathbf{y}_\perp - \mathbf{Y}_\perp^{(1)} \bar{\boldsymbol{\beta}}^{(1)}\|_{\ell_2}^2 \leq C^2 \sigma^2$. ■

7.3.2 Proof of Theorem 5.6.3

First, Lemma 7.3.2 asserts that by using τ and $f(t)$ as stated, our choice of λ obeys

$$\lambda > \frac{\sigma}{\sqrt{8d_1}}. \quad (7.3.17)$$

All we need is to demonstrate that when λ is as above, there are no false discoveries. To do this, it is sufficient to establish that the solution $\hat{\boldsymbol{\beta}}^{(1)}$ to the *reduced problem*

obeys

$$\left\| \mathbf{Y}^{(\ell)T} (\mathbf{y} - \mathbf{Y}^{(1)} \hat{\boldsymbol{\beta}}^{(1)}) \right\|_{\ell_\infty} < \lambda, \quad \text{for all } \ell \neq 1. \quad (7.3.18)$$

This is a consequence of this:

Lemma 7.3.6 Fix $\mathbf{A} \in \mathbb{R}^{d \times N}$ and $T \subset \{1, 2, \dots, N\}$. Suppose that there is a solution \mathbf{x}^* to

$$\min \frac{1}{2} \|\mathbf{y} - \mathbf{Ax}\|_{\ell_2}^2 + \lambda \|\mathbf{x}\|_{\ell_1} \quad \text{subject to} \quad \mathbf{x}_{T^c} = \mathbf{0}$$

obeying $\|\mathbf{A}_{T^c}^T (\mathbf{y} - \mathbf{Ax}^*)\|_{\ell_\infty} < \lambda$. Then any optimal solution $\hat{\mathbf{x}}$ to

$$\min \frac{1}{2} \|\mathbf{y} - \mathbf{Ax}\|_{\ell_2}^2 + \lambda \|\mathbf{x}\|_{\ell_1}$$

must also satisfy $\hat{\mathbf{x}}_{T^c} = \mathbf{0}$.

Proof Consider a perturbation $\mathbf{x}^* + t\mathbf{h}$. For $t > 0$ sufficiently small, the value of the LASSO functional at this point is equal to

$$\begin{aligned} \frac{1}{2} \|\mathbf{y} - \mathbf{A}(\mathbf{x}^* + t\mathbf{h})\|_{\ell_2}^2 + \lambda \|\mathbf{x} + t\mathbf{h}\|_{\ell_1} &= \frac{1}{2} \|\mathbf{y} - \mathbf{Ax}^*\|_{\ell_2}^2 - t \langle \mathbf{A}^T (\mathbf{y} - \mathbf{Ax}^*), \mathbf{h} \rangle + \frac{t^2}{2} \|\mathbf{Ah}\|_{\ell_2}^2 \\ &\quad + \lambda \|\mathbf{x}\|_{\ell_1} + \lambda t \langle \text{sgn}(\mathbf{x}_T), \mathbf{h}_T \rangle + \lambda t \|\mathbf{h}_{T^c}\|_{\ell_1}. \end{aligned}$$

Now since the optimality conditions give that $\mathbf{A}_T^T (\mathbf{y} - \mathbf{Ax}^*) = \lambda \text{sgn}(\mathbf{x}_T)$ and that by assumption, $\mathbf{A}_{T^c}^T (\mathbf{y} - \mathbf{Ax}^*) = \lambda \epsilon_{T^c}$ with $\|\epsilon_{T^c}\|_{\ell_\infty} < 1$, the value of the LASSO functional is equal to

$$\frac{1}{2} \|\mathbf{y} - \mathbf{Ax}^*\|_{\ell_2}^2 + \lambda \|\mathbf{x}\|_{\ell_1} + \frac{t^2}{2} \|\mathbf{Ah}\|_{\ell_2}^2 + \lambda t (\|\mathbf{h}_{T^c}\|_{\ell_1} - \langle \epsilon_{T^c}, \mathbf{h}_{T^c} \rangle).$$

Clearly, when $\mathbf{h}_{T^c} \neq 0$, the value at $\mathbf{x}^* + t\mathbf{h}$ is strictly greater than that at \mathbf{x}^* , which proves the claim. \blacksquare

We return to (7.3.18) and write

$$\begin{aligned} \mathbf{Y}^{(\ell)T} (\mathbf{y} - \mathbf{Y}^{(1)} \hat{\boldsymbol{\beta}}^{(1)}) &= \mathbf{X}^{(\ell)T} (\mathbf{y}_\parallel - \mathbf{Y}_\parallel^{(1)} \hat{\boldsymbol{\beta}}^{(1)}) + \mathbf{X}^{(\ell)T} (\mathbf{y}_\perp - \mathbf{Y}_\perp^{(1)} \hat{\boldsymbol{\beta}}^{(1)}) \\ &\quad + \mathbf{Z}^{(\ell)T} (\mathbf{y}_\parallel - \mathbf{Y}_\parallel^{(1)} \hat{\boldsymbol{\beta}}^{(1)}) + \mathbf{Z}^{(\ell)T} (\mathbf{y}_\perp - \mathbf{Y}_\perp^{(1)} \hat{\boldsymbol{\beta}}^{(1)}). \end{aligned}$$

To establish (7.3.18), we shall control the ℓ_∞ norm of each term by using Lemma 7.3.1 and the estimates concerning the size of the residuals. For ease of presentation we assume $d_1 \geq d_\ell$ and $d_\ell \leq n - d_1$ —the proof when $d_\ell > d_1$ is similar.

The term $\mathbf{X}^{(\ell)T}(\mathbf{y}_\parallel - \mathbf{Y}_\parallel^{(1)}\hat{\boldsymbol{\beta}}^{(1)})$. Using Lemma 7.3.1 with $a = 2\sqrt{\log N}$, $b = 2\sqrt{2\log N}$, we have

$$\left\| \mathbf{X}^{(\ell)T}(\mathbf{y}_\parallel - \mathbf{Y}_\parallel^{(1)}\hat{\boldsymbol{\beta}}^{(1)}) \right\|_{\ell_\infty} \leq \sqrt{32} \log N \frac{\text{aff}(S_1, \S_\ell)}{\sqrt{d_1}} \left\| \mathbf{y}_\parallel - \mathbf{Y}_\parallel^{(1)}\hat{\boldsymbol{\beta}}^{(1)} \right\|_{\ell_2};$$

this holds uniformly over $\ell \neq 1$ with probability at least $1 - \frac{4}{N^2}$. Now applying Lemma 7.3.5 we conclude that

$$\left\| \mathbf{X}^{(\ell)T}(\mathbf{y}_\parallel - \mathbf{Y}_\parallel^{(1)}\hat{\boldsymbol{\beta}}^{(1)}) \right\|_{\ell_\infty} \leq \lambda L_N \text{aff}(S_1, S_\ell) := \lambda I_1.$$

The term $\mathbf{X}^{(\ell)T}(\mathbf{y}_\perp - \mathbf{Y}_\perp^{(1)}\hat{\boldsymbol{\beta}}^{(1)})$. As before,

$$\left\| \mathbf{X}^{(\ell)T}(\mathbf{y}_\perp - \mathbf{Y}_\perp^{(1)}\hat{\boldsymbol{\beta}}^{(1)}) \right\|_{\ell_\infty} \leq \sqrt{32} \log N \frac{1}{\sqrt{n-d_1}} \left\| \mathbf{y}_\perp - \mathbf{Y}_\perp^{(1)}\hat{\boldsymbol{\beta}}^{(1)} \right\|_{\ell_2},$$

which holds uniformly over $\ell \neq 1$ with probability at least $1 - \frac{4}{N^2}$ (we used the fact that the affinity is at most one.) Applying Lemma 7.3.5 gives

$$\left\| \mathbf{X}^{(\ell)T}(\mathbf{y}_\perp - \mathbf{Y}_\perp^{(1)}\hat{\boldsymbol{\beta}}^{(1)}) \right\|_{\ell_\infty} \leq L_N \frac{\sigma}{\sqrt{n}} := I_2.$$

The terms $\mathbf{Z}^{(\ell)T}(\mathbf{y}_\parallel - \mathbf{Y}_\parallel^{(1)}\hat{\boldsymbol{\beta}}^{(1)})$ and $\mathbf{Z}^{(\ell)T}(\mathbf{y}_\perp - \mathbf{Y}_\perp^{(1)}\hat{\boldsymbol{\beta}}^{(1)})$. Since $\mathbf{Z}^{(\ell)}$ is a Gaussian matrix with entries $\mathcal{N}(0, \sigma^2/n)$, applying Lemma B.0.12 gives

$$\begin{aligned} \left\| \mathbf{Z}^{(\ell)T}(\mathbf{y}_\parallel - \mathbf{Y}_\parallel^{(1)}\hat{\boldsymbol{\beta}}^{(1)}) \right\|_{\ell_\infty} &\leq 2\sigma \sqrt{\frac{2\log N}{n}} \left\| \mathbf{y}_\parallel - \mathbf{Y}_\parallel^{(1)}\hat{\boldsymbol{\beta}}^{(1)} \right\|_{\ell_2} \\ &\leq C\lambda\sigma \sqrt{\frac{d_1 \log N}{n}} := \lambda I_3. \end{aligned}$$

with probability at least $1 - \frac{2}{N^2}$

In a similar fashion with probability at least $1 - \frac{2}{N^2}$, we have

$$\left\| \mathbf{Z}^{(\ell)}^T (\mathbf{y}_\perp - \mathbf{Y}_\perp^{(1)} \hat{\boldsymbol{\beta}}^{(1)}) \right\|_{\ell_\infty} \leq \sigma^2 \sqrt{\frac{L_N}{n}} := I_4.$$

Putting all this together, we need

$$(I_1 + I_3)\lambda + I_2 + I_4 < \lambda$$

to hold with high probability. It is easy to see that the affinity condition in Theorem 5.6.3 is equivalent to $I_1 + I_3 < 1 - \frac{1}{\sqrt{3}}$. Therefore it suffices to have $\lambda > \sqrt{3}(I_2 + I_4)$.

The latter holds if

$$\lambda > L_N \frac{\sigma}{\sqrt{n}}.$$

The calculations have been performed assuming (7.3.17). Therefore, it suffices to have

$$\lambda > \sqrt{\frac{\sigma}{8d_1}} \max\left(1, L_N \sqrt{\frac{d_1}{n}}\right).$$

The simplifying assumption $d_1 \leq n/L_N^2$ at the beginning of the paper concludes the proof.

7.3.3 The size of the solution to the projected problem

This section proves Lemma 7.3.3. We begin with a definition.

Definition 7.3.7 (Restricted isometry property (RIP)) *We say that $\mathbf{A} \in \mathbb{R}^{d \times N}$ obeys $RIP(s, \delta)$ if*

$$(1 - \delta) \|\mathbf{x}\|_{\ell_2} \leq \|\mathbf{Ax}\|_{\ell_2} \leq (1 + \delta) \|\mathbf{x}\|_{\ell_2}$$

holds for all s -sparse vectors (vectors such that $\|\mathbf{x}\|_{\ell_0} \leq s$).

We mentioned that Lemma 7.3.3 is essentially contained in the work of Wojtaszczyk and now make this clear. Below, $\hat{\mathbf{x}}$ is any optimal solution to

$$\min \|\mathbf{x}\|_{\ell_1} \quad \text{subject to} \quad \mathbf{y} = \mathbf{Ax}. \tag{7.3.19}$$

Theorem 7.3.8 [244, Theorem 3.4] Suppose $\mathbf{A} \in \mathbb{R}^{d \times N}$ obeys RIP(s, δ) and $r(\mathcal{P}(\mathbf{A})) \geq \frac{\mu}{\sqrt{s}}$, where $\mathcal{P}(\mathbf{A})$ is the symmetrized convex hull of the columns of \mathbf{A} . Then there is a universal constant $C = C(\delta, \mu)$, such that for any solution \mathbf{x} to $\mathbf{y} = \mathbf{A}\mathbf{x}$, we have

$$\|\hat{\mathbf{x}} - \mathbf{x}\|_{\ell_2} \leq C \|\mathbf{x} - \mathbf{x}_{(s)}\|_{\ell_2} + C \|\mathbf{y} - \mathbf{A}\mathbf{x}_{(s)}\|_{\ell_2}. \quad (7.3.20)$$

Above, $\mathbf{x}_{(s)}$ is the best s -sparse approximation to \mathbf{x} (the vector \mathbf{x} with all but the s -largest entries set to zero).

We now prove Lemma 7.3.3 and begin with $\lambda = 0$. The expected squared Euclidean norm of a column of $\mathbf{Y}_{\parallel}^{(1)}$ is equal to $1 + \sigma^2 d/n$. Rescaling $\mathbf{Y}_{\parallel}^{(1)}$ as $\mathbf{A} = (1 + \sigma^2 d/n)^{-1/2} \mathbf{Y}_{\parallel}^{(1)}$, it is a simple calculation to show that with $s = \sqrt{d_1/L_{N_1/d_1}}$ (recall that L_{N_1/d_1} is a constant times $\log(N_1/d_1)$), \mathbf{A} obeys RIP(s, δ) for a fixed numerical constant δ ; see Section 5.6.1 in [232]. For the same value of s , a simple rescaling of Lemma C.0.18 asserts that as long as $\rho \geq \rho^*$, $r(\mathcal{P}(\mathbf{A})) \geq \mu/\sqrt{s}$ for a fixed numerical constant μ .

Now let \mathbf{A}^\dagger be the pseudo-inverse of \mathbf{A} , and set $\boldsymbol{\beta} = \mathbf{A}^\dagger \mathbf{y}_{\parallel}$. First, $\|\mathbf{y}_{\parallel}\|_{\ell_2} \in [3/4, 5/4]$ and second $\|\boldsymbol{\beta}\|_{\ell_2} \leq 1$ as shown in Lemma 7.3.2. Thus,

$$\|\tilde{\boldsymbol{\beta}}^{(1)} - \boldsymbol{\beta}\|_{\ell_2} \leq C \|\boldsymbol{\beta}\|_{\ell_2} + C \|\mathbf{y}_{\parallel} - \mathbf{A}\boldsymbol{\beta}_{(s)}\|_{\ell_2} \leq \frac{9}{4}C + C(1 + \delta).$$

The second inequality comes from the RIP property $\|\mathbf{A}\boldsymbol{\beta}_{(s)}\|_{\ell_2} \leq (1 + \delta) \|\boldsymbol{\beta}_{(s)}\|_{\ell_2} \leq (1 + \delta)$. This completes the proof for $\lambda = 0$. For $\lambda > 0$, one simply applies the same argument with \mathbf{y}_{\parallel} replaced by $\mathbf{Y}_{\parallel}^{(1)} \tilde{\boldsymbol{\beta}}^{(1)}$.

7.3.4 Proof of Theorem 5.6.4

Once we know there are no false discoveries, the many many-true-discovery result becomes quite intuitive. The reason is this: we already know that the residual sum of squares $\|\mathbf{y}_{\parallel} - \mathbf{Y}_{\parallel}^{(1)} \hat{\boldsymbol{\beta}}^{(1)}\|_{\ell_2}^2$ is much smaller than one provided λ is not too large—this is why we have the upper bound $f(t) \leq \alpha_0/t$. Now recall that \mathbf{y}_{\parallel} is a generic

point in a d -dimensional subspace and, therefore, it cannot be well approximated as a short linear combination of points taken from the same subspace. It is possible—and not difficult—to make this mathematically precise and thus prove Theorem 5.6.4. Here, we take a shorter route re-using much of what we have already seen and/or established.

We rescale $\mathbf{Y}_{\parallel}^{(1)}$ as $\mathbf{A} = (1 + \sigma^2 d/n)^{-1/2} \mathbf{Y}_{\parallel}^{(1)}$ to ensure that the expected squared Euclidean norm of each column is one, and set $s = \sqrt{d_1/L_{N_1/d_1}}$. We introduce three events E_1 , E_2 and E_3 .

- E_1 : there are no false discoveries.
- E_2 : with s as above, the two conditions in Theorem 7.3.8 hold.
- $E_3 = \{\|\hat{\beta}^{(1)}\|_{\ell_1} \geq c_1 \sqrt{d_1/\log \rho_1}\}$ for some numerical constant c_1 .

Since $f(t) \geq \sigma/(\sqrt{2}t)$, there are no false discoveries with high probability. Further, in the proof of this theorem we established that E_1 and E_2 hold with high probability. Last, E_3 also has large probability.

Lemma 7.3.9 *Let $f(t)$ be as in Theorem 5.6.4, then the event E_3 has probability at least $1 - e^{-\gamma_3 d_1} - \frac{10}{N}$.*

Proof The proof is nearly the same as that of the lower bound in Lemma 7.3.2.³ By optimality of $\hat{\beta}^{(1)}$ for (7.3.1), $\hat{\beta}^{(1)}$ is also a solution to

$$\min \|\beta\|_{\ell_1} \quad \text{subject to} \quad \|\mathbf{y} - \mathbf{Y}^{(1)}\beta\|_{\ell_2} \leq \tau$$

with $\tau = \|\mathbf{y} - \mathbf{Y}^{(1)}\hat{\beta}^{(1)}\|_{\ell_2}$ (in this optimization problem we regard τ as fixed). Hence, if we can take τ to be sufficiently smaller than one, then the argument in the proof of Lemma 7.3.2 can be copied to establish the claim.

³Notice that the properties needed for this lemma to hold are the same as that of Lemma 7.3.2 and, therefore, E_3 holds under the conditions of the no-false discovery theorem. Hence, the guaranteed probabilities of success of our two main theorems are the same.

Moving forward, Lemma 7.3.5 gives

$$\begin{aligned}\|\mathbf{y} - \mathbf{Y}^{(1)}\hat{\boldsymbol{\beta}}^{(1)}\|_{\ell_2} &\leq \frac{32}{3} \lambda \sqrt{\frac{d_1}{\log \rho_1}} + C\sigma \\ &\leq \frac{32}{3} \frac{\alpha_0}{\text{Val(Step 1)}} \sqrt{\frac{d_1}{\log \rho_1}} + C\sigma \\ &\leq 110\alpha_0 + C\sigma.\end{aligned}$$

The second inequality follows from the definition of the regularization parameter since $\lambda \leq \alpha_0/\text{Val(Step 1)}$ and the third from (7.3.3) in Lemma 7.3.2. We have proved the claim provided α_0 and σ are sufficiently small (this is the place where the assumption about σ comes into play). \blacksquare

Lemma 7.3.10 *On the event $E_2 \cap \{\|\hat{\boldsymbol{\beta}}^{(1)}\|_{\ell_0} \leq s\}$, where s is as above,*

$$\|\hat{\boldsymbol{\beta}}^{(1)}\|_{\ell_2} \leq c_2$$

for some numerical constant c_2 .

Proof We apply Theorem 7.3.8 one more time with \mathbf{A} as before, $\mathbf{y} = (1 + \sigma^2 d/n)^{-1/2} \mathbf{y}_{\parallel}$, $\hat{\mathbf{x}} = \bar{\boldsymbol{\beta}}^{(1)}$ and $\mathbf{x} = \hat{\boldsymbol{\beta}}^{(1)}$. By assumption, $\mathbf{x}_{(s)} = \mathbf{x}$, and

$$\|\bar{\boldsymbol{\beta}}^{(1)} - \hat{\boldsymbol{\beta}}^{(1)}\|_{\ell_2} \leq C(1 + \sigma^2 d/n)^{-1/2} \|\mathbf{y}_{\parallel} - \mathbf{Y}_{\parallel}^{(1)} \hat{\boldsymbol{\beta}}^{(1)}\|_{\ell_2} \leq C$$

(the second inequality comes from Lemma 7.3.5). The proof follows by applying the triangular inequality and Lemma 7.3.3 with $\lambda = 0$. \blacksquare

The proof of Theorem 5.6.4 is now straightforward. With s as above, there is nothing to do if $\|\hat{\boldsymbol{\beta}}^{(1)}\|_{\ell_0} \geq s$, so we assume $\|\hat{\boldsymbol{\beta}}^{(1)}\|_{\ell_0} \leq s$. On this event and $E_1 \cap E_2 \cap E_3$,

$$\frac{\|\hat{\boldsymbol{\beta}}^{(1)}\|_{\ell_1}}{\|\hat{\boldsymbol{\beta}}^{(1)}\|_{\ell_2}} \geq c_3 \sqrt{\frac{d_1}{\log \rho_1}}.$$

Cauchy-Schwartz asserts that $\|\hat{\beta}^{(1)}\|_{\ell_1} \leq \sqrt{\|\hat{\beta}^{(1)}\|_{\ell_0}} \|\hat{\beta}^{(1)}\|_{\ell_2}$ and, therefore,

$$\|\hat{\beta}^{(1)}\|_0 \geq Cd_1/\log\rho_1.$$

7.4 Proof of results with gross outliers

Our proofs rely heavily on techniques from Geometric Functional Analysis and we now introduce some basic concepts and results from this field. Most of our exposition is adapted from [231].

7.4.1 Background on Geometric Functional Analysis

Definition 7.4.1 *The maximal and average values of $\|\cdot\|_{\mathcal{K}}$ on the sphere S^{n-1} are defined by*

$$b(\mathcal{K}) = \sup_{\mathbf{x} \in S^{n-1}} \|\mathbf{x}\|_{\mathcal{K}} \quad \text{and} \quad M(\mathcal{K}) = \int_{S^{n-1}} \|\mathbf{x}\|_{\mathcal{K}} d\sigma(\mathbf{x}).$$

Above, σ is the uniform probability measure on the sphere.

Definition 7.4.2 *The mean width $M^*(\mathcal{K})$ of a symmetric convex body \mathcal{K} in \mathbb{R}^n is the expected value of the dual norm over the unit sphere,*

$$M^*(\mathcal{K}) = M(\mathcal{K}^\circ) = \int_{S^{n-1}} \|\mathbf{y}\|_{\mathcal{K}^\circ} d\sigma(\mathbf{y}) = \int_{S^{n-1}} \max_{\mathbf{z} \in \mathcal{K}} \langle \mathbf{y}, \mathbf{z} \rangle d\sigma(\mathbf{y}).$$

With this in place, we now record some useful results.

Lemma 7.4.3 *We always have $M(\mathcal{K})M(\mathcal{K}^\circ) \geq 1$.*

Proof Observe that since $\|\cdot\|_{\mathcal{K}^\circ}$ is the dual norm of $\|\cdot\|_{\mathcal{K}}$, $\|\mathbf{x}\|^2 = \|\mathbf{x}\|_{\mathcal{K}} \|\mathbf{x}\|_{\mathcal{K}^\circ}$ and thus

$$1 = \left(\int_{S^{n-1}} \sqrt{\|\mathbf{x}\|_{\mathcal{K}} \|\mathbf{x}\|_{\mathcal{K}^\circ}} d\sigma \right)^2 \leq \int_{S^{n-1}} \|\mathbf{x}\|_{\mathcal{K}} d\sigma \int_{S^{n-1}} \|\mathbf{x}\|_{\mathcal{K}^\circ} d\sigma,$$

where the inequality follows from Cauchy-Schwarz. ■

The following theorem deals with concentration properties of norms. According to [143], these appear in the first pages of [178].

Theorem 7.4.4 (Concentration of measure) *For each $t > 0$, we have*

$$\sigma\left\{\mathbf{x} \in S^{n-1} : \left|\|\mathbf{x}\|_{\mathcal{K}} - M(\mathcal{K})\right| > tM(\mathcal{K})\right\} < \exp\left(-ct^2 n \left[\frac{M(\mathcal{K})}{b(\mathcal{K})}\right]^2\right),$$

where $c > 0$ is a universal constant.

The following lemma is a simple modification of a well-known result in Geometric Functional Analysis.

Lemma 7.4.5 (Many faces of convex symmetric polytopes) *Let \mathcal{P} be a symmetric polytope with f faces. Then*

$$n \left(\frac{M(\mathcal{P})}{b(\mathcal{P})} \right)^2 \leq c \log(f),$$

for some positive numerical constant $c > 0$.

Definition 7.4.6 (Geometric Banach-Mazur Distance) *Let \mathcal{K} and \mathcal{L} be symmetric convex bodies in \mathbb{R}^n . The Banach-Mazur distance between \mathcal{K} and \mathcal{L} , denoted by $d(\mathcal{K}, \mathcal{L})$, is the least positive value $ab \in \mathbb{R}$ for which there is a linear image $T(\mathcal{K})$ of \mathcal{K} obeying*

$$b^{-1}\mathcal{L} \subseteq T(\mathcal{K}) \subseteq a\mathcal{L}.$$

Theorem 7.4.7 (John's Theorem) *Let \mathcal{K} be a symmetric convex body in \mathbb{R}^n and B_2^n be the unit ball of \mathbb{R}^n . Then $d(\mathcal{K}, B_2^n) \leq \sqrt{n}$.*

Our proofs make use of two theorems concerning volume ratios. The first is this.

Lemma 7.4.8 (Urysohn's inequality) *Let $\mathcal{K} \subset \mathbb{R}^n$ be a compact set. Then*

$$\left(\frac{\text{vol}(\mathcal{K})}{\text{vol}(B_2^n)} \right)^{\frac{1}{n}} \leq M^*(\mathcal{K}).$$

Lemma 7.4.9 [24, Theorem 2] Let $\mathcal{K}^o = \{\mathbf{z} \in \mathbb{R}^n : |\langle \mathbf{a}_i, \mathbf{z} \rangle| \leq 1 : i = 1, \dots, N\}$ with $\|\mathbf{a}_i\|_{\ell_2} = 1$. The volume of \mathcal{K}^o admits the lower estimate

$$\text{vol}(\mathcal{K}^o)^{1/n} \geq \begin{cases} \frac{2\sqrt{2}}{\sqrt{p}r}, & p \geq 2, \\ \frac{1}{r}, & \text{if } 1 \leq p \leq 2. \end{cases}$$

Here, $n \leq N$, $1 \leq p < \infty$ and $r = \left(\frac{1}{n} \sum_{i=1}^N \|\mathbf{a}_i\|_{\ell_2}^p \right)^{\frac{1}{p}}$.

7.4.2 Proof of Theorem 5.5.2

We begin with two lemmas relating the mean and maximal value of norms with respect to convex polytopes.

Lemma 7.4.10 For a symmetric convex body in \mathbb{R}^n ,

$$\frac{M(\mathcal{K})M(\mathcal{K}^o)}{b(\mathcal{K})b(\mathcal{K}^o)} \geq \frac{1}{\sqrt{n}}.$$

Proof Variants of this lemma are well known in geometric functional analysis. By definition,

$$\|x\|_{\mathcal{K}} \leq b(\mathcal{K})\|x\|_2,$$

$$\|x\|_{\mathcal{K}^o} \leq b(\mathcal{K}^o)\|x\|_2,$$

and, hence, the property of dual norms allows us to conclude that

$$\begin{aligned} \frac{1}{b(\mathcal{K}^o)}\|x\|_2 &\leq \|x\|_{\mathcal{K}} \leq b(\mathcal{K})\|x\|_2, \\ \frac{1}{b(\mathcal{K})}\|x\|_2 &\leq \|x\|_{\mathcal{K}^o} \leq b(\mathcal{K}^o)\|x\|_2. \end{aligned}$$

However, using Definition 7.4.6, these relationships imply that $d(\mathcal{K}, B_2^n) = b(\mathcal{K})b(\mathcal{K}^o)$. Therefore,

$$\frac{M(\mathcal{K})M(\mathcal{K}^o)}{b(\mathcal{K})b(\mathcal{K}^o)} = \frac{M(\mathcal{K})M(\mathcal{K}^o)}{d(\mathcal{K}, B_2^n)}.$$

Applying John's lemma and using Lemma 7.4.3 concludes the proof. ■

Lemma 7.4.11 *For a convex symmetric polytope $\mathcal{K}(\mathbf{A})$, $\mathbf{A} \in \mathbb{R}^{n \times N}$, we have*

$$n \left(\frac{M(\mathcal{K})}{b(\mathcal{K})} \right)^2 \geq c \frac{n}{\log(2N)}.$$

Proof By Lemma 7.4.10, we know that

$$\frac{M(\mathcal{K})M(\mathcal{K}^o)}{b(\mathcal{K})b(\mathcal{K}^o)} \geq \frac{1}{\sqrt{n}} \quad \Rightarrow \quad \frac{M(\mathcal{K})}{b(\mathcal{K})} \geq \frac{1}{\sqrt{n} \frac{M(\mathcal{K}^o)}{b(\mathcal{K}^o)}}.$$

However, applying Lemma 7.4.5 to the polytope \mathcal{K}^o , which has $2N$ faces, gives

$$n \left(\frac{M(\mathcal{K}^o)}{b(\mathcal{K}^o)} \right)^2 \leq C \log(2N) \quad \Rightarrow \quad \frac{1}{\sqrt{n} \frac{M(\mathcal{K}^o)}{b(\mathcal{K}^o)}} \geq \frac{1}{\sqrt{C \log(2N)}}.$$

These two inequalities imply

$$\frac{M(\mathcal{K})}{b(\mathcal{K})} \geq \frac{1}{\sqrt{C \log(2N)}} \quad \Rightarrow \quad n \left(\frac{M(\mathcal{K})}{b(\mathcal{K})} \right)^2 \geq \frac{1}{C} \frac{n}{\log(2N)}.$$

■

7.4.2.1 Proof of Theorem 5.5.2 (part (a))

The proof is in two steps.

1- For every inlier point $\mathbf{x}_i^{(\ell)}$,

$$\text{optval}(\mathbf{x}_i^{(\ell)}, \mathbf{X}_{(-i)}) \leq \frac{1}{r(\mathcal{P}_{-i}^\ell)}.$$

2- For every outlier point $\mathbf{x}_i^{(0)}$, with probability at least $1 - e^{-c \frac{nt^2}{\log N}}$, we have

$$(1-t) \frac{\vartheta(\rho)}{\sqrt{e}} \sqrt{n} \leq \text{optval}(\mathbf{x}_i^{(0)}, \mathbf{X}_{(-i)}).$$

7.4.2.1.1 Proof of step 1

Lemma 7.4.12 Suppose $\mathbf{y} \in \text{Range}(\mathbf{A})$, then

$$\text{optval}(\mathbf{y}, \mathbf{A}) \leq \frac{\|\mathbf{y}\|_{\ell_2}}{r(\mathcal{K}(\mathbf{A}))}.$$

Proof As stated before,

$$\text{optval}(\mathbf{y}, \mathbf{A}) = \|\mathbf{y}\|_{\mathcal{K}(\mathbf{A})}.$$

Put $\mathcal{K}(\mathbf{A}) = \mathcal{K}$ for short. Using the definition of the max norm and circumradius

$$\|\mathbf{y}\|_{\mathcal{K}} = \|\mathbf{y}\|_{\ell_2} \left\| \frac{\mathbf{y}}{\|\mathbf{y}\|_{\ell_2}} \right\|_{\mathcal{K}} \leq \|\mathbf{y}\|_{\ell_2} b(\mathcal{K}) = \|\mathbf{y}\|_{\ell_2} R(\mathcal{K}^o) = \frac{\|\mathbf{y}\|_{\ell_2}}{r(\mathcal{K})}. \quad (7.4.1)$$

The last equality follows from the fact that maximal norm on the unit sphere and the inradius are the inverse of one another (Lemma 7.2.3). ■

Notice that

$$\text{optval}(\mathbf{x}_i^{(\ell)}, \mathbf{X}_{(-i)}) \leq \text{optval}(\mathbf{x}_i^{(\ell)}, \mathbf{X}_{(-i)}^{(\ell)}).$$

and since $\|\mathbf{x}_i^{(\ell)}\|_{\ell_2} = 1$, applying the above lemma with $\mathbf{y} = \mathbf{x}_i^{(\ell)}$ and $\mathbf{A} = \mathbf{X}_{(-i)}^{(\ell)}$, gives

$$\text{optval}(\mathbf{x}_i^{(\ell)}, \mathbf{X}_{(-i)}^{(\ell)}) \leq \frac{1}{r(\mathcal{P}_{-i}^{\ell})}.$$

Combining these two identities establishes (7.4.1).

7.4.2.1.2 Proof of step 2

We are interested in lower bounding $\text{optval}(\mathbf{y}, \mathbf{A})$ in which \mathbf{A} is a fixed matrix and $\mathbf{y} \in \mathbb{R}^n$ is chosen uniformly at random on the unit sphere. Our strategy consists in finding a lower bound in expectation, and then using a concentration argument to derive a bound that holds with high probability.

Lemma 7.4.13 (Lower bound in expectation) Suppose $\mathbf{y} \in \mathbb{R}^n$ is a point chosen uniformly at random on the unit sphere and $\mathbf{A} \in \mathbb{R}^{n \times N}$ is a matrix with unit-norm columns. Then

$$\mathbb{E}\{\text{optval}(\mathbf{y}, \mathbf{A})\} > \begin{cases} \frac{1}{\sqrt{e}} \sqrt{\frac{2}{\pi}} \frac{n}{\sqrt{N}}, & \text{if } 1 \leq \frac{N}{n} \leq e, \\ \frac{1}{\sqrt{e}} \sqrt{\frac{2}{\pi e}} \sqrt{\frac{n}{\log \frac{N}{n}}}, & \text{if } \frac{N}{n} \geq e. \end{cases}$$

Proof Since $\text{optval}(\mathbf{y}, \mathbf{A}) = \|\mathbf{y}\|_{\mathcal{K}(\mathbf{A})}$, the expected value is equal to $M^*(\mathcal{K}^o) = M(\mathcal{K})$. Applying Urysohn's Theorem (Theorem 7.4.8) gives

$$M^*(\mathcal{K}^o) \geq \left(\frac{\text{vol}(\mathcal{K}^o)}{\text{vol}(B_2^n)} \right)^{\frac{1}{n}}.$$

It is well known that the volume of the n -dimensional sphere with radius one is given by

$$\text{vol}(B_2^n) = \frac{\pi^{n/2}}{\Gamma(\frac{n}{2} + 1)}.$$

The well-known Stirling approximation gives

$$\Gamma\left(\frac{n}{2} + 1\right) \geq \sqrt{2\pi} e^{-n/2} \left(\frac{n}{2}\right)^{(n+1)/2},$$

and, therefore, the volume obeys

$$\text{vol}(B_2^n) \leq \left(\sqrt{\frac{2\pi e}{n}} \right)^n.$$

Note that if $\{\mathbf{a}_i\}_{i=1}^N$ is a family of n -dimensional unit-norm vectors, then for $p \geq 1$,

$$\left(\frac{1}{n} \sum_{i=1}^N |\mathbf{a}_i|^p \right)^{\frac{1}{p}} \leq \left(\frac{N}{n} \right)^{\frac{1}{p}}.$$

Applying Lemma 7.4.9 for $p \geq 2$ gives

$$\text{vol}(\mathcal{K}^o)^{\frac{1}{n}} \geq \frac{2\sqrt{2}}{\sqrt{p}\left(\frac{N}{n}\right)^{\frac{1}{p}}}.$$

The right-hand side is maximum when $p = 2 \log \frac{N}{n}$, which is larger than 2 as long as $\frac{N}{n} \geq e$. When $\frac{N}{n} < e$, we shall use $p = 2$. Plugging in this value of p in the bound of Lemma 7.4.9, we conclude that

$$\text{vol}(\mathcal{K}^o)^{\frac{1}{n}} \geq \begin{cases} \frac{2}{\sqrt{\frac{N}{n}}}, & \text{if } 1 \leq \frac{N}{n} \leq e, \\ \frac{2}{\sqrt{e}} \frac{1}{\sqrt{\log \frac{N}{n}}}, & \text{if } \frac{N}{n} \geq e. \end{cases}.$$

Finally, this identity together with the approximation of the volume of the sphere conclude the proof. \blacksquare

Lemma 7.4.14 (Concentration around mean) *In the setup of Lemma 7.4.13,*

$$\text{optval}(\mathbf{y}, \mathbf{A}) \geq (1-t)\mathbb{E}\{\text{optval}(\mathbf{y}, \mathbf{A})\},$$

with probability at least $1 - e^{-c \frac{nt^2}{\log(2N)}}$.

Proof The proof follows from Theorem 7.4.4 and applying Lemma 7.4.11. \blacksquare

These two lemmas (Lower bound in expected value and Concentration around mean), combined with the union bound give the first part of Theorem 5.5.2.

7.4.2.2 Proof of Theorem 5.5.2 part (b)

This part follows from the combination of the proof of Theorem 5.5.2 part (a) with the bound given for the inradius presented in the proof of Theorem 5.3.6.

7.4.3 Proof of Theorem 5.5.1

The proof follows Theorem 5.5.2 with t a small number. Here we use $t = 1 - \frac{1}{\sqrt{2}}$.

7.5 Proof of results with missing data

For the sake of simplicity in this section we focus on one optimization problem of the form (4.6.4). We assume that \mathbf{x} is a point from S_1 and the rest of the data points are arranged as columns of a matrix \mathbf{X} . We use Ω to denote the index of the entries revealed to us from point \mathbf{x} . Similarly, we use Ω_j to denote the index of the entries we get to observe from the j th column of \mathbf{X} . As we described earlier we set

$$\widehat{\boldsymbol{\Gamma}} = \mathbf{Y}_\Omega^T \mathbf{Y}_\Omega - \delta \text{diag}(\mathbf{Y}_\Omega^T \mathbf{Y}_\Omega), \quad \widehat{\boldsymbol{\gamma}} = \mathbf{Y}_\Omega^T \mathbf{x}_\Omega,$$

and solve problems of the form

$$\min \|\boldsymbol{\beta}\|_{\ell_1} \quad \text{subject to} \quad \|\widehat{\boldsymbol{\gamma}} - \widehat{\boldsymbol{\Gamma}}\boldsymbol{\beta}\|_{\ell_\infty} \leq \lambda. \quad (7.5.1)$$

We also remind the reader that

$$\boldsymbol{\beta}^I = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^N} \|\boldsymbol{\beta}\|_{\ell_1} \quad \text{subject to} \quad \mathbf{X}_\Omega \boldsymbol{\beta} = \mathbf{x}_\Omega. \quad (7.5.2)$$

7.5.1 Proof of Theorem 5.6.2

Define $\Xi_\Omega = \mathbf{Y}_\Omega - \mathbf{X}_\Omega$ and $\mathbf{D} = \delta \text{diag}(\mathbf{Y}_\Omega^T \mathbf{Y}_\Omega)$. Since $\mathbf{y}_\Omega = \mathbf{x}_\Omega = \mathbf{X}_\Omega \boldsymbol{\beta}^I$,

$$\mathbf{Y}_\Omega^T (\mathbf{y}_\Omega - \mathbf{Y}_\Omega \boldsymbol{\beta}^I) + \mathbf{D} \boldsymbol{\beta}^I = \Xi_\Omega^T \mathbf{x}_\Omega + \left(\mathbf{X}_\Omega^T \mathbf{X}_\Omega - \mathbf{Y}_\Omega^T \mathbf{Y}_\Omega + \mathbf{D} \right) \boldsymbol{\beta}^I$$

Define the matrix \mathbf{Q}

$$Q_{ij} = \begin{cases} (1 - \delta)^2 & \text{if } i \neq j \\ (1 - \delta) & \text{if } i = j \end{cases}$$

where δ'_i are i.i.d. Bernoulli(δ) random variables. Let \oslash denote entry wise division and set $\mathbf{w} = \Xi_\Omega^T \mathbf{x}_\Omega$ and $\tilde{\boldsymbol{\Gamma}} = \tilde{\mathbf{Y}}_\Omega^T \tilde{\mathbf{Y}}_\Omega \oslash \mathcal{P}_\Omega \mathbf{Q} \mathcal{P}_\Omega - \mathbf{X}_\Omega^T \mathbf{X}_\Omega$, we have

$$\|\mathbf{Y}_\Omega^T (\mathbf{y}_\Omega - \mathbf{Y}_\Omega \boldsymbol{\beta}^I) + \mathbf{D} \boldsymbol{\beta}^I\|_{\ell_\infty} \leq \|\mathbf{w}\|_{\ell_\infty} + \|\tilde{\boldsymbol{\Gamma}} \boldsymbol{\beta}^I\|_{\ell_\infty}.$$

the result follows by bounding each of the above terms.

Bounding $\|\mathbf{w}\|_{\ell_\infty}$

Notice that it has mean zero and the j th entry can be written as

$$\mathbf{w}_j = \frac{1}{(1-\delta)} \sum_{i=1}^n (\delta - \delta'_i)(1 - \delta_i) X_{ij} x_i.$$

Let $\mathbf{x} = \mathbf{U}\mathbf{a}$ with \mathbf{U} is the basis of the subspace \mathbf{x} belongs to. Similarly, set $\mathbf{x}_j = \mathbf{V}\mathbf{b}_j$, be the j -column of \mathbf{X} where \mathbf{V} is the basis of the subspace \mathbf{x}_j belongs to (potentially $\mathbf{V} = \mathbf{U}$). Notice that $X_{ij}x_i = \mathbf{a}^T \mathbf{u}_i \mathbf{v}_i^T \mathbf{b}_j$. We have

$$\begin{aligned} \mathbb{E}_{\delta, \delta'} \{ w_j^2 \} &= \frac{1}{(1-\delta)^2} \sum_{i=1}^n \mathbb{E}_{\delta_i, \delta'_i} \{ (\delta - \delta'_i)^2 (1 - \delta_i)^2 \} (\mathbf{a}^T \mathbf{u}_i \mathbf{v}_i^T \mathbf{b}_j)^2 \\ &= \delta \sum_{i=1}^n (\mathbf{a}^T \mathbf{u}_i \mathbf{v}_i^T \mathbf{b}_j)^2, \\ &= \delta M_j. \end{aligned}$$

Therefore, applying a weighted Hoeffding inequality we have

$$\mathbb{P}_{\delta, \delta'} \{ |w_j| > t \} \leq e^{-\frac{t^2}{2\delta M_j}}.$$

Set $M_{ij} = \mathbf{a}^T \mathbf{u}_i \mathbf{v}_i^T \mathbf{b}_j$, we have $M_j = \sum_{i=1}^n M_{ij}^2$. It follows from Lemma 7.3.1 that for all $i = 1, 2, \dots, n$ and $j = 1, 2, \dots, N$ with probability $1 - 4/(nN)$ we have

$$\begin{aligned} |M_{ij}| &\leq \sqrt{8} \log(nN) \frac{\|\mathbf{u}_i \mathbf{v}_i^T\|_F}{\sqrt{d_1} \sqrt{d_2}}, \\ &\leq \sqrt{8} \log(nN) \frac{\|\mathbf{u}_i\|_{\ell_2} \|\mathbf{v}_i^T\|_{\ell_2}}{\sqrt{d_1} \sqrt{d_2}}, \\ &\leq \sqrt{8} \log(nN) \frac{\mu^2}{n}. \end{aligned}$$

Thus $M_j \leq \sqrt{8} \log(nN) \frac{\mu^2}{\sqrt{n}}$. Therefore, applying the union bound we conclude that with probability at least $1 - 1/N - 4/(nN)$

$$\|\mathbf{w}\|_{\ell_\infty} \leq 4\mu^2 \log(nN) \frac{\sqrt{\delta \log N}}{\sqrt{n}}.$$

Bounding $\|\tilde{\Gamma}\beta^I\|_{\ell_\infty}$

We first bound the absolute value of each of the entries of $\tilde{\Gamma}$ denoted by $\|\tilde{\Gamma}\|_{\ell_\infty}$.

Notice that

$$\begin{aligned}\tilde{\Gamma}_{ii} &= \frac{1}{(1-\delta)} \sum_{k=1}^n (\delta - \delta'_k)(1 - \delta_k) X_{ki}^2, \\ \tilde{\Gamma}_{ij} &= \frac{1}{(1-\delta)^2} \sum_{k=1}^n [2\delta - \delta^2 - \delta_{k'} - \delta_{k''} + \delta_{k'}\delta_{k''}] (1 - \delta_k) X_{ki} X_{kj}.\end{aligned}$$

Applying the same calculation we did for the bound on $\|\mathbf{w}\|_{\ell_\infty}$ we have that for $i = 1, 2, \dots, N$ with probability $1 - 1/N - 4/(nN)$ we have

$$|\tilde{\Gamma}_{ii}| \leq 4\mu^2 \log(nN) \frac{\sqrt{\delta \log N}}{\sqrt{n}}. \quad (7.5.3)$$

Set $X_i = \mathbf{U}\mathbf{a}_i$ and $X_j = \mathbf{V}\mathbf{b}_j$, with \mathbf{U} and \mathbf{V} denoting the basis of the subspaces the i -th and j -th points belong to. We have

$$\begin{aligned}\mathbb{E}_{\delta, \delta', \delta''} \{ \tilde{\Gamma}_{ij}^2 \} &= \frac{1}{(1-\delta)^4} \sum_{k=1}^n \mathbb{E}_{\delta, \delta', \delta''} \{ [2\delta - \delta^2 - \delta_{k'} - \delta_{k''} + \delta_{k'}\delta_{k''}]^2 (1 - \delta_k)^2 \} (\mathbf{a}^T \mathbf{u}_k \mathbf{v}_k^T \mathbf{b}_j)^2, \\ &= \frac{\delta(2-\delta)}{(1-\delta)} M, \\ &\leq 2\delta M.\end{aligned}$$

Now following calculations paralleling what was done for bounding $\|\mathbf{w}\|_{\ell_\infty}$ for all $i \neq j$ with probability $1 - 1/N - 4/n$

$$|\tilde{\Gamma}_{ij}| \leq 8\mu^2 \log(nN) \frac{\sqrt{\delta \log N}}{\sqrt{n}}. \quad (7.5.4)$$

Combining (7.5.3) and (7.5.4) we conclude that with probability at least $1 - 2/n - 8/N$

$$\|\tilde{\Gamma}\|_{\ell_\infty} \leq 8\mu^2 \log(nN) \frac{\sqrt{\delta \log N}}{\sqrt{n}}. \quad (7.5.5)$$

We will use $\tilde{\Gamma}_i$ to denote the i -th row of $\tilde{\Gamma}$. Note that

$$\tilde{\Gamma}_i \boldsymbol{\beta}^I = \sum_{j=1}^N \tilde{\Gamma}_{ij} (\boldsymbol{\beta}^I)_j = \sum_{j=1}^N \epsilon_j |\tilde{\Gamma}_{ij}| |(\boldsymbol{\beta}^I)_j|,$$

where $\epsilon_j = \text{sgn}(\tilde{\Gamma}_{ij} (\boldsymbol{\beta}^I)_j)$. Note that ϵ_j are i.i.d. ± 1 Radamacher random variables independent of \mathbf{x}_Ω and \mathbf{X}_Ω and thus also of $\boldsymbol{\beta}^I$. Therefore applying a weighted Hoeffding inequality we have

$$\mathbb{P}\{|\tilde{\Gamma}_i \boldsymbol{\beta}^I| > t \mid \mathbf{x}_\Omega, \mathbf{X}_\Omega\} \leq e^{-\frac{t^2}{2M'}}.$$

Here, $M' = \sum_{j=1}^N (\tilde{\Gamma}_{ij} (\boldsymbol{\beta}^I)_j)^2$. Applying the union bound we conclude that

$$\|\tilde{\Gamma} \boldsymbol{\beta}^I\|_{\ell_\infty} \leq \sqrt{2(\log N) M'} \leq \sqrt{2 \log N} \|\tilde{\Gamma}\|_{\ell_\infty} \|\boldsymbol{\beta}^I\|_{\ell_2},$$

holds with probability at least $1 - 2/N$. Combining (7.5.5) and Lemma 7.5.1 below we conclude that

$$\|\tilde{\Gamma} \boldsymbol{\beta}^I\|_{\ell_\infty} \leq C \frac{(\log N)^2}{\sqrt{n}} \sqrt{\frac{\delta}{1-\delta}},$$

holds with high probability for a fixed numerical constant C .

Lemma 7.5.1 *Assume that $\rho_1 \geq N_1/d_1$. Then there exists an absolute constant C such that*

$$\|\boldsymbol{\beta}^I\|_{\ell_2} \leq C \sqrt{1-\delta},$$

holds with probability at least $1 - 5e^{-\gamma_1 d_1} - e^{-\sqrt{N_1 d_1}}$.

Proof The basic steps of the proof are similar to Lemma 7.3.8 except that we need to lower bound the inradius of the symmetrized convex hull of the columns of $\mathbf{X}_\Omega^{(1)}$ (denoted by $r(\mathbf{X}_\Omega^{(1)})$) instead of $r(\mathbf{X}^{(1)})$. It is easy to see that

$$r(\mathcal{P}(\mathbf{X}_\Omega^{(1)})) \geq r(\mathcal{P}(\mathbf{X}^{(1)})) \sigma_{\min}(\mathcal{P}_\Omega \mathbf{U}^{(1)}).$$

So we need to lower bound the minimum eigenvalue of $\mathcal{P}_\Omega \mathbf{U}^{(1)}$. Define \mathbf{u}_i as the i th

row of $\mathbf{U}^{(1)}$ and notice that

$$\mathbf{W} = \mathbf{U}^{(1)T} \mathcal{P}_\Omega \mathbf{U}^{(1)} = \sum_{i=1}^n \delta_i \mathbf{u}_i \mathbf{u}_i^T = \sum_{i=1}^n \mathbf{W}_i,$$

where δ_i is a random variable which is 1 with probability $1 - \delta$ and 0 with probability δ . By the matrix Chernoff bound we have

$$\mathbb{P}\{\sigma_{\min}(\mathbf{W}) \leq t\sigma_{\min}(\mathbb{E}\mathbf{W})\} \leq 2d_1 e^{-\frac{(1-t)^2 \sigma_{\min}(\mathbb{E}\mathbf{W})}{2R}},$$

where $R = \max_i \|\mathbf{u}_i\|_{\ell_2}^2$. Notice that $\mathbb{E}\mathbf{W} = (1 - \delta)\mathbf{I}$ and $\|\mathbf{u}_i\|_{\ell_2}^2 \leq \mu^2 \frac{d_1}{n}$. If we want the probability of success to be at least $1 - 1/N^2$ using $t = 1/2$ we need to have

$$1 - \delta \geq 8\mu^2 \frac{d_1}{n} (\log 2d_1 + 2 \log N)$$

Putting all of this together we have that as long as we have the following bound on the ratio of missing entries

$$\delta \leq 1 - 8\mu^2 \frac{d_1}{n} (\log 2d_1 + 2 \log N)$$

with high probability we have

$$r(\mathcal{P}(\mathbf{X}_\Omega^{(1)})) \geq r(\mathcal{P}(\mathbf{X}^{(1)})) \sqrt{\frac{1-\delta}{2}} \geq \frac{1}{4\sqrt{2}} \sqrt{\frac{\log \rho_1}{d_1}} \sqrt{1-\delta}.$$

The last inequality follows from Lemma 7.2.4. ■

Part II

Phase Retrieval

Chapter 8

The generalized phase retrieval problem

In many areas of science and engineering, we only have access to magnitude measurements as detectors can often only record the modulus of the scattered radiation from an object and not its phase. Imagine then that we have a discrete object $\mathbf{x} \in \mathbb{C}^n$, and that we would like to measure $\langle \mathbf{a}_r, \mathbf{x} \rangle$ for some sampling vectors $\mathbf{a}_r \in \mathbb{C}^n$ but only have access to phaseless measurements of the form

$$y_r = |\langle \mathbf{a}_r, \mathbf{x} \rangle|^2, \quad r = 1, \dots, m. \quad (8.0.1)$$

The generalized phase retrieval problem is that of recovering the missing phase of the data $\langle \mathbf{a}_k, \mathbf{x} \rangle$. Once we know the phase one can easily find the vector \mathbf{x} by essentially solving a system of linear equations. More specifically, we are interested in solving quadratic equations of the form

$$y_r = |\langle \mathbf{a}_r, \mathbf{z} \rangle|^2, \quad r = 1, 2, \dots, m, \quad (8.0.2)$$

where $\mathbf{z} \in \mathbb{C}^n$ is the decision variable, $\mathbf{a}_r \in \mathbb{C}^n$ are known sampling vectors, and $y_r \in \mathbb{R}$ are observed measurements.

The quintessential phase retrieval problem, or phase problem for short, asks to recover a signal from the modulus of its Fourier transform. This comes from the fact

that in coherent X-ray imaging, it follows from the Fraunhofer diffraction equation that the optical field at the detector is well approximated by the Fourier transform of the object of interest. Since photographic plates, CCDs and other light detectors can only measure light intensity, the problem is then to recover $\mathbf{x} = \{x[t]\}_{t=0}^{n-1} \in \mathbb{C}^n$ from measurements of the type

$$y_r = \left| \sum_{t=0}^{n-1} x[t] e^{-i2\pi\omega_r t} \right|^2, \quad \omega_r \in \Omega, \quad (8.0.3)$$

where Ω is a sampled set of frequencies in $[0, 1]$ (we stated the problem in one dimension to simplify matters). We thus recognize an instance of (8.0.1) in which the vectors \mathbf{a}_r are sampled values of complex sinusoids. X-ray diffraction images are of this form, and as is well known, permitted the discovery of the double helix [239]. In addition to X-ray crystallography [123, 177], the phase problem has numerous other applications in the imaging sciences such as diffraction and array imaging [49, 69], optics [238], speckle imaging in astronomy [81], and microscopy [174]. Other areas where related problems appear include acoustics [18, 21], blind channel estimation in wireless communications [8, 203], interferometry [83], quantum mechanics [78, 205] and quantum information [129].

Because of the practical significance of the phase retrieval problem in imaging science, over the past century numerous heuristics have been developed for its solution; mostly for the special case where one samples the (square) modulus of the Fourier transform of the signal as in (8.0.3). We shall review some of these approaches in greater detail in Chapter 10. The most popular algorithms of this kind were derived from the pioneering research of Gerchberg and Saxton [114] and Fienup [103, 105]. The Gerchberg-Saxton algorithm starts from a random initial estimate and proceeds by iteratively applying a pair of ‘projections’: at each iteration, the current guess is projected in data space so that the magnitude of its frequency spectrum matches the observations; the signal is then projected in signal space to conform to some a-priori knowledge about its structure. In a typical instance, our knowledge may be that the signal is real-valued, nonnegative and spatially limited. First, while these methods often work well in practice, the algorithms seem to rely heavily on a priori information

about the signals, see [104, 106, 126, 210]. Second, since these algorithms can be cast as alternating projections onto nonconvex sets [30] (the constraint in Fourier space is not convex), fundamental mathematical questions concerning their convergence remain, for the most part, unresolved.

On the theoretical side, multiple papers study the uniqueness (up to global phase) of the solution to the generalized phase retrieval problem. More specifically, they study how many measurements are required for the mapping described in (8.0.1) to be injective. We shall review some of these results in Chapter 10. However, these theories often do not explain when and how it is possible to get to this unique solution using a tractable algorithm.

A more recent line of work shows that algorithms based on classical convex relaxations for solving quadratic equations (known as Schor's semidefinite relaxations) may provide a tractable solution to the generalized phase retrieval problem. More specifically, the papers [55, 64] show that if the sampling vectors \mathbf{a}_r are sufficiently randomized, then it is possible to use tractable Semi-Definite Programs (SDPs) to recover the unknown signal (up to global phase) using a near minimal number of observations. Even though these results are rather interesting from a theoretical point of view, their practical value is less clear. First, the random models studied in these papers are very far from the kind of data one can collect in an actual experiment such as X -ray imaging. Second, and more importantly, while SDP based relaxations offer "tractable" solutions they become computationally prohibitive as the dimension of the signal increases. Indeed, the memory requirements of these SDP based relaxations renders them impractical for large signal sizes such as modestly sized images (Please see Section 14.3 for further detail).

In this part of this dissertation we wish to address the important challenges mentioned above and close the huge gap that currently exists between the theory and practice of phase retrieval. Our goal is to arrive at algorithms that are not only provably correct in a physically realizable setting but also easy to implement and effective on real images. To this aim we shall first review some of the applications of phase retrieval and existing theory and algorithms in Chapters 9 and 10. Then in Chapter 11, we show that SDP based algorithms are provably effective in a physically inspired

setup where one can modulate the signal of interest and then let diffraction occur. Furthermore, we show that these algorithms are stable vis a vis noise. While, this result gives us a formal proof of a tractable phase retrieval algorithm in a physically realizable setup, due to the prohibitive nature of the SDPs involved, it is still far from practical. To address this issue in Chapter 12 we develop a new approach to the phase retrieval problem based on non-convex optimization. We show that our algorithm allows the exact retrieval of phase information from a nearly minimal number of random measurements. Furthermore, we rigorously show that our scheme is efficient both in terms of computational and data resources. For example, a variation on this scheme leads to a near-linear time algorithm for a physically realizable model based on coded diffraction patterns. Finally, we prove that our algorithm is also near optimal in terms of stability in the presence of noise. Complimenting our theoretical study, in Chapter 14 we illustrate the effectiveness of our methods with various experiments on synthetic and image data. Finally, in Chapter 13 we show that classical algorithms of Gerchberg-Saxton and Fienup can also be viewed from the lens of non-convex optimization and sketch theoretical results towards rigorous proofs of their convergence.

The material in this part of the dissertation are based on the papers [57, 58] and yet unpublished notes by the author. In particular the noiseless results of Chapters 11 and 12 with their proofs in Chapter 15 are based on these papers. This also applies to most parts of our numerical results in Chapter 14. We also note that the text of these parts also follows the text of the corresponding papers closely. The results of Chapter 13 as well as the stability of both the convex and non-convex schemes in Chapters 11 and 12 (along with their proofs and corresponding numerical simulations in Chapters 14 and 15) are based on yet unpublished notes by the author.

Chapter 9

Applications of phase retrieval

We review some applications of phase retrieval in this chapter. Our aim is to explain how the Fourier phase retrieval problem—recovery of a signal given the magnitude of its Fourier transform—arises naturally in multiple applications.

9.1 Applications in optical imaging

Perhaps the most widespread use of algorithmic phase retrieval is in optical imaging. In optical imaging scientists constantly wish to increase resolution so as to be able to image smaller and smaller details. This in turn requires imaging at shorter and shorter wavelengths. The problem is that lens-like devices and other optical components are very difficult (essentially impossible) to make at very short wavelengths. Algorithmic phase retrieval offers an alternative method for recovery of phase structure without the need for involved measuring setups such as mirrors and lenses. In the ensuing paragraphs we shall provide a brief historical overview of phase retrieval in optical imaging followed by a description of how it comes about in Coherent Diffraction Imaging (CDI).¹ CDI is currently a very active area where one of the main challenges is 3D structural determination of large protein complexes [172, 188]. We focus on this

¹Please see [215] for a more comprehensive read on applications of phase retrieval to optical imaging. We also emphasize that our description of the applications in optical imaging is influenced by this tutorial.

application as this is one of the areas where algorithmic phase retrieval can potentially have a tremendous impact.

9.1.1 Some history

In an insightful paper [211] from 1952, Sayre envisioned the use of phase retrieval in X-ray diffraction microscopy. A few decades later, in 1978, Fienup demonstrated empirically the recovery of phase information about 2D images from the magnitude of their Fourier transform, using additional knowledge such as non-negativity and known support [103]. Around the turn of the millennium as researchers in optical imaging started experimenting with new X-ray sources, there was a revival in the use of phase retrieval in optical imaging. This revival was mainly driven by the need for lensless imaging. This line of research gained a lot of traction due to the first experimental recording and reconstruction of a diffraction pattern of a non-crystalline object by Miao and collaborators in 1999 [174] which gave way to Coherent Diffraction Imaging (CDI). Since then, these techniques have been used successfully to image a variety of samples, ranging from nano-particles, nano-crystals, and biomaterials to whole cells [70, 71, 93, 111, 163, 168, 170, 171, 173, 175, 176, 190, 191, 199, 200, 206, 214, 222, 242, 243, 246, 254]. The hope is that these techniques will eventually lead to successful imaging of large protein complexes and biological specimens.

9.1.2 Coherent Diffraction Imaging (CDI)

In a basic CDI setup a light source with a quasi-monochromatic coherent wave sheds light on the object and then the diffracted intensity is measured far away from the object. This setup is depicted in Figure 9.1. For the sake of exposition we shall assume that the direction of the light is parallel to the z-axis and the object and measurement (diffraction) planes are parallel to the xy-plane with the object at $z = 0$ and the image plane at $z = d$. Let $E_{in}(x, y)$ and $E_{out}(x, y)$ be the electrical field at the

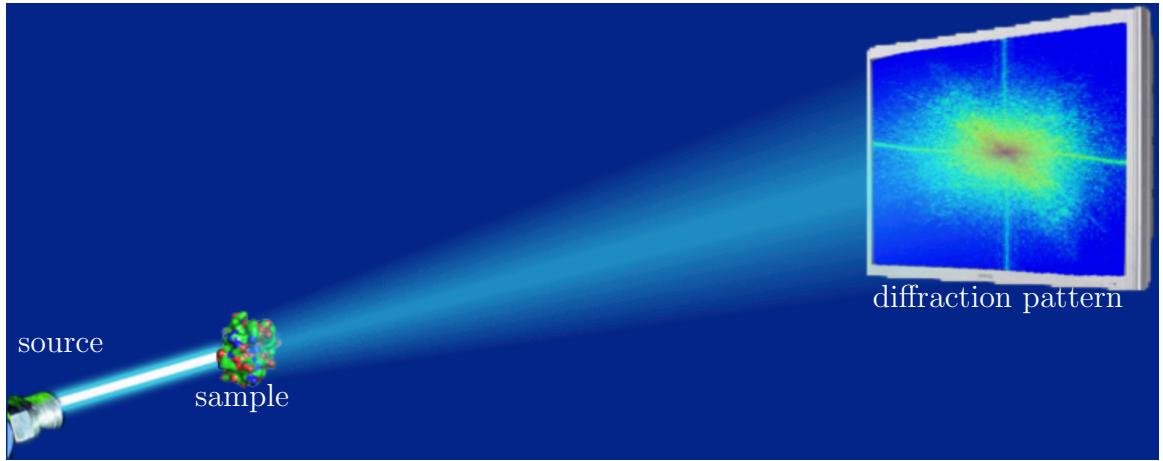


Figure 9.1: An illustrative setup for Coherent Diffraction Imaging: A coherent wave diffracts from a sample, and produces a far-field diffraction pattern which corresponds to the magnitude of the Fourier transform of the sample.

object and image planes.² Using the Huygens-Fresnel Principle³ $E_{out}(x, y)$ is related to $E_{in}(x, y)$ via

$$E_{out}(x, y) = \frac{i}{\lambda d} e^{-i\frac{2\pi}{\lambda}d} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} E_{in}(x', y') \cdot \exp\left(-i\pi \frac{(x-x')^2 + (y-y')^2}{\lambda d}\right) dx' dy', \quad (9.1.1)$$

where λ is the wavelength of the monochromatic light source. The phase in the argument of the exponent is $(\pi/\lambda d)[(x-x')^2 + (y-y')^2] = (\pi/\lambda d)[(x^2 + y^2) + (x'^2 + y'^2) - 2(xx' + yy')]$. Assuming that the object is confined to a small area of radius b and if the distance d is sufficiently large so that $b^2/\lambda d$ is small, then the phase factor $(\pi/\lambda d)(x'^2 + y'^2) \leq \pi(b^2/\lambda d)$ is negligible and (9.1.1) is approximately given by

$$E_{out}(x, y) \approx \frac{i}{\lambda d} e^{-i\frac{2\pi}{\lambda}d} \cdot \exp\left(-i\pi \frac{x^2 + y^2}{\lambda d}\right) \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} E_{in}(x', y') \cdot \exp\left(-i2\pi \frac{xx' + yy'}{\lambda d}\right) dx' dy'.$$

²We note that $E_{in}(x, y)$ is the 2D image we wish to recover and is proportional to the line integral of the electron density of the object along the z direction. We shall explain this connection in greater detail in Section 14.4.

³Please see [117] page 52 for further details.

Let \hat{E}_{in} denote the 2D Fourier transform of E_{in} we have

$$E_{out}(x, y) \approx \frac{i}{\lambda d} e^{-i \frac{2\pi}{\lambda} d} \cdot \exp\left(-i\pi \frac{x^2 + y^2}{\lambda d}\right) \hat{E}_{in}\left(\frac{x}{\lambda d}, \frac{y}{\lambda d}\right). \quad (9.1.2)$$

Assuming that we also limit ourselves to points in the image place within a radius α centered about the z -axis so that $\pi(x^2 + y^2)/(\lambda d) \leq \pi\alpha^2/(\lambda d) \ll \pi$ the phase factor $\exp[-i\pi(x^2 + y^2)/\lambda d]$ in (9.1.2) can also be ignored. Therefore, assuming $a^2/(\lambda d) \ll 1$ and $b^2/(\lambda d) \ll 1$ which is known as the Fraunhofer approximation we have

$$I_{out}(x, y) = |E_{out}(x, y)|^2 \propto \left| \hat{E}_{in}\left(\frac{x}{\lambda d}, \frac{y}{\lambda d}\right) \right|^2.$$

Here, $I_{out}(x, y)$ is the measured intensity. This far field intensity is measured, and the goal is to recover E_{in} . We shall see in Section 14.4 that recovering E_{in} is equivalent to recovering the object. Therefore, in CDI we face a continuous instance of the phase retrieval problem.

9.2 Speckle imaging in astronomy

Another interesting application of phase retrieval is in Astronomy. An important task in astronomy is to get accurate images of extra terrestrial objects such as stars through the telescope. However, turbulence in the atmosphere causes a point of light such as a star to appear distorted and unfocused through a telescope on earth. Astronomers have developed numerous techniques to mitigate the resolution downgrade that occurs due to atmospheric distortion. One popular approach is through speckle interferometry which was developed by Labeyrie in 1970.

Let $i(x, y)$ represent the observed two-dimensional image and let $o(x, y)$ represent the corresponding object. Also let $h(x, y)$ denote the combined telescope-atmosphere point-spread function which describes the light distribution when a point source is imaged by the telescope. We shall use $\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix}$ to denote the spatial variable. We

have

$$i(\mathbf{x}) = o(\mathbf{x}) * h(\mathbf{x}), \quad (9.2.1)$$

where $*$ denotes two-dimensional convolution. Here, $\boldsymbol{\nu}$ represents the two-dimensional spatial-frequency variable corresponding to the spatial variable \mathbf{x} and also $I(\boldsymbol{\nu})$, $O(\boldsymbol{\nu})$, and $H(\boldsymbol{\nu})$ are the two-dimensional Fourier transforms of $i(\mathbf{x})$, $o(\mathbf{x})$, and $h(\mathbf{x})$. In Fourier space we have

$$I(\boldsymbol{\nu}) = O(\boldsymbol{\nu}) \cdot H(\boldsymbol{\nu}).$$

The dilemma faced by Labeyrie was that even though a highly magnified image of a point source such as a star could be obtained with a brief photographic exposure, this high magnification spread the light on his detector so that there was little total exposure. Labeyrie realized that a single exposure that froze the turbulence was not sufficient, so he decided to use additional exposures. The problem was that new exposures produced different samples of the turbulence and averaging many exposures directly would still contain the distortion due to turbulence and still resulted in a bad image. Therefore, Labeyrie focused his attention on the power spectra of the frames. Let $i^{(r)}(\mathbf{x})$ for $r = 1, 2, \dots, m$ denote the images from different exposures (known as speckles) and $h^{(r)}(\mathbf{x})$ the different samples of the turbulence. Figure 9.2 shows an example of these speckles along with the target object. Note that the object is assumed to be constant during the different exposures. Thus, using the model of (9.2.1) we have

$$I^{(r)}(\boldsymbol{\nu}) = O(\boldsymbol{\nu}) \cdot H^{(r)}(\boldsymbol{\nu}).$$

Therefore, averaging over the different samples we have

$$\frac{1}{m} \sum_{r=1}^m |I^{(r)}(\boldsymbol{\nu})|^2 = |O(\boldsymbol{\nu})|^2 \left(\frac{1}{m} \sum_{r=1}^m |H^{(r)}(\boldsymbol{\nu})|^2 \right). \quad (9.2.2)$$

The reason the averaging strategy is effective is that atmospherical turbulence typ-

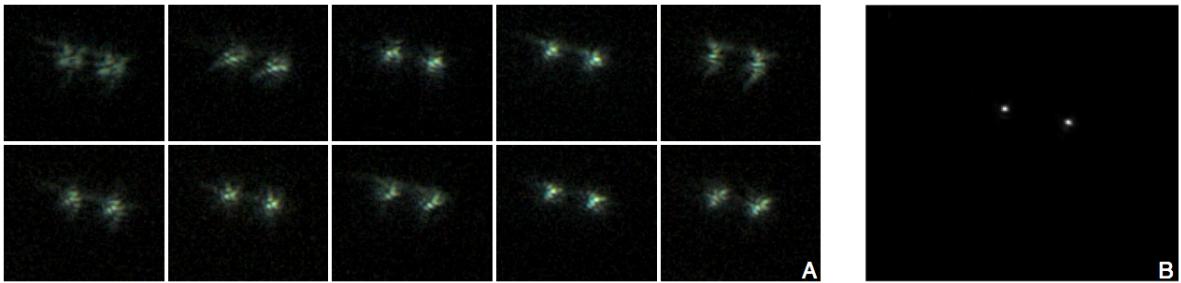


Figure 9.2: An example of phase retrieval for speckle imaging. In (A) we see 10 sample speckle images of a double star (ϵ Lyrae). In (B) we see the high resolution image of the same star obtained through phase retrieval techniques. This figure is from [130, 203].

ically only affect the phase of the object and not its magnitude. This is due to the well established fact in astronomy that by considering exposure times below 10ms the turbulence in the atmosphere can be treated as a frozen pattern of phase variations. Returning our attention to (9.2.2) the left-hand side is the average of the power spectrum of the different measured images and is known. The second term in the right-hand side can be estimated by a separate measurement of a point source (such as a star) under similar seeing conditions. Therefore, (9.2.2) allows us to estimate the power spectral of the object fairly accurately. Now given this power spectra the goal is to recover the object $o(x, y)$, which is an instance of the Fourier phase retrieval problem.

9.3 Blind channel estimation

An interesting application of phase retrieval is in the estimation of a communication channel.⁴ Estimating the channel response function is an essential part of wireless communication systems [28, 203] such as Orthogonal Frequency Division Multiplexing (OFDM). We use $f(t)$ and $h(t)$ to denote the input signal and channel response

⁴Oddly enough the author used to work on OFDM channel estimation during undergrad. So this is somehow full circle!

functions. In this case the output is given by

$$y(t) = (h * f)(t). \quad (9.3.1)$$

Here, the input signal is typically the result of frequency modulation using a discrete sequence. We do not know the input. We have access to samples from the channel and wish to estimate both the input signal and channel response using these samples. However, we do have some information about the input. The input signal is usually whitened to achieve the maximum channel capacity. That is,

$$\mathbb{E}|\hat{f}(\nu)|^2 = c \quad \text{for all } \nu, \quad (9.3.2)$$

where \hat{f} denotes the Fourier transform of signal. Here, the expectation is with respect to the randomness in the signal. Looking at (9.3.1) in the frequency domain together with (9.3.2) we have

$$\hat{y}(\nu) = \hat{f}(\nu) \cdot \hat{h}(\nu) \quad \Rightarrow \quad |\hat{h}(\nu)|^2 \propto \mathbb{E}|\hat{y}(\nu)|^2.$$

Since we have access to samples from the output we can estimate $\mathbb{E}|\hat{y}(\nu)|^2$ fairly accurately using its empirical average. This in turn implies that we can estimate $|\hat{h}(\nu)|^2$ up to a constant rather accurately. Again, we arrive at an instance of the phase retrieval problem where we wish to recover the signal $h(t)$ from the magnitude of its Fourier transform $|\hat{h}(\nu)|^2$. The reason that estimating the channel h is important is that using our estimate of the channel we can in turn estimate the signal f and therefore recover the input sequence.

Chapter 10

Prior art in phase retrieval

In this chapter we will discuss some of the classic algorithms and models for phase retrieval. We shall also review results which explain when the generalized phase retrieval problem in (8.0.1) has a unique solution.

10.1 When is the phase retrieval problem unique?

In this section we shall review results from the literature which give conditions under which the phase retrieval problem has a unique solution (up to a global phase factor). In the process we shall also become familiar with the different models for the sensing vectors \mathbf{a}_r that have been studied in the literature. For this purpose, let \mathbf{A} be the $m \times n$ matrix whose r th row is \mathbf{a}_r^* so that with obvious notation equation (8.0.1) takes the form $\mathbf{y} = |\mathbf{Ax}|^2$. We study when the mapping $\mathbf{z} \mapsto |\mathbf{Az}|^2$ is injective.

10.1.1 Fourier measurements

As we discussed in Chapter 9, one of the most common forms of phase retrieval is when we wish to recover a signal from the magnitude of its Fourier transform. These measurements in the discrete case take the form

$$y_r = \left| \sum_{t=0}^{n-1} x[t] e^{-i\omega_r t} \right|^2, \quad \text{with } \omega_r = \frac{2\pi r}{n} \quad \text{for } r = 1, 2, \dots, n. \quad (10.1.1)$$

Throughout this part of this dissertation by Fourier measurements we mean the above form. In this case the sensing matrix $\mathbf{A} = \mathbf{F}$, where \mathbf{F} denotes the $n \times n$ DFT matrix. First we note that there are trivial ambiguities in the Fourier phase retrieval problem as global phase shift ($x[n] \Rightarrow x[n] \cdot e^{j\theta}$), conjugate inversion ($x[n] \Rightarrow \overline{x[-n]}$), and spatial shift ($x[n] \Rightarrow \overline{x[n+n_0]}$) do not affect the Fourier magnitude, so one can only hope to recover the signal up to these ambiguities. However, even beyond these ambiguities unique recovery is not possible as one could assign any phase to the magnitudes measurements and take an inverse Fourier transform to arrive at a different solution that obeys the magnitude constraints $\mathbf{y} = |\mathbf{F}\mathbf{x}|^2$. Therefore, for the Fourier phase retrieval problem to be solvable (up to the ambiguities) we either need to take additional measurements or assume some apriori structure about the signal. We shall briefly review some results of this nature below.

- **bounded support.** One common assumption is that the support of the signal \mathbf{x} is bounded. While this assumption does not resolve the uniqueness issue for 1D signals, interestingly it is useful for signals of dimensions two and higher. Bruck and Sodin [48], Hayes [126], and Bates [29] showed that except for a set of signals of measure zero, a real valued signal with compact support in dimensions two and higher is uniquely specified by the magnitude of its continuous Fourier transform (up to the ambiguities discussed above).
- **sparsity.** An often convenient structural assumption is that the signal $\mathbf{x} \in \mathbb{C}^n$ is sparse in the sense that it contains only a small number—say s —nonzero element with $s \ll n$. In this case it might be even possible to use less than n Fourier measurements i.e. only use a fraction of the measurements of (10.1.1). Indeed, [203] building on prior work shows that when $s \neq 6$ under some technical assumptions on the support of the signal $m \geq s^2 - s + 1$ measurements are sufficient to guarantee uniqueness. These results can also identify the set of signals for which this uniqueness does not occur based on whether the z-transform of the signal is factorable or not. Please see [126] for further detail.
- **OversampleD DFT.** Another interesting approach to overcome the uniqueness issue is to try to have additional measurements in some form. A common

approach is to sample the continuous problem at a finer scale. That is, use $\omega_r = 2\pi r/m$ with $r = 1, 2, \dots, m$ with $m \geq n$. Based on the argument that the autocorrelation function of any object is twice the size of the object itself, Bates [29] in 1982 concluded that for real-valued signals phases can be uniquely retrieved from the intensity if and only if a diffraction pattern is over sampled by a factor of two [174], i.e. $m \geq 2n - 1$.

10.1.2 General measurements

In this section we shall focus on the uniqueness of the generalized phase retrieval problem. As stated earlier we wish to study the injectivity of the mapping $\mathbf{z} \rightarrow |\mathbf{A}\mathbf{z}|^2$ up to a global phase factor. Our exposition is inspired by a fantastic blog post [2] on this subject. To facilitate our exposition we shall define $m_{\min}(n)$ to be the smallest number of intensity measurements of an n dimensional signal for which the mapping is injective. More specifically, there exists $m_{\min}(n)$ sensing vectors \mathbf{a}_r for which the above mapping is injective and there is no ensemble of measurement vectors of smaller size.

Let us first focus on the injectivity of the mapping $\mathbf{z} \rightarrow |\mathbf{A}\mathbf{z}|^2$ when both \mathbf{z} and \mathbf{A} are real valued. In this case Balan, Casazza and Edidin in [21] established an equivalent characterization of injectivity (referred to as the complement property in the literature). They used this characterization to show that $m_{\min}(n) = 2n - 1$. Indeed the results of [21] was stronger in the sense that not only did they show that if $m < 2n - 1$ then injectivity is impossible but in addition they showed that the mapping is injective for almost all sensing matrices as long as $m \geq 2n - 1$. To give an example, Gaussian sensing vectors are injective in this case with probability one.

Turning our attention to the complex case while a variation of the complement property was established in [26] a precise characterization of $m_{\min}(n)$ does not exist. However, the culmination of 26 years of research [21, 44, 107, 235, 240] has narrowed down this value to the interval $4n - 2H(n-1) - 3 \leq m_{\min}(n) \leq 4n - 4$. Here, $H(n-1) \leq \log n$ is the number of 1's in the binary expansion of $n - 1$. Indeed, similar to the real case the upper bound on the interval holds in a stronger sense. That is, the mapping

is injective for many sensing matrices when $m \geq 4n - 4$. We shall end this section with a formal statement of this result which has short but rather elegant proof due to [77].

Theorem 10.1.1 [77] *If $m \geq 4n - 4$, then for a generic sensing matrix \mathbf{A} the mapping $\mathbf{z} \rightarrow |\mathbf{Az}|^2$ from \mathbb{C}^n to \mathbb{R}^m is injective.*

The meaning of generic here is that \mathbf{A} corresponds to a point in a non-empty Zariski open subset of $\mathbb{C}^{m \times n}$. In particular, what this theorem tells us is that when $m \geq 4n - 4$, there is an open dense set of sensing matrices \mathbf{A} (in the Euclidean topology of $\mathbb{C}^{m \times n}$) for which the mapping is injective. Please see [77] for further details.

10.2 Classical approaches to phase retrieval

In this section we shall explain some of the classical approaches to phase retrieval. Our presentation is by no means comprehensive and only covers the error reduction algorithm followed by a few of its variations.

10.2.1 Error reduction algorithm

Earlier, we discussed the Error Reduction (ER) algorithm due to Gerchberg-Saxton and Fienup. These formulations were originally developed for particular sensing matrices \mathbf{A} . We shall discuss the form this algorithm takes for the generalized phase retrieval problem and then briefly mention these particular instances.

These algorithms can be described as follows: suppose \mathbf{z}_τ is the current guess, then one computes the image of \mathbf{z}_τ through \mathbf{A} and adjust its modulus so that it matches that of the observed data vector: with obvious notation,

$$\hat{\mathbf{v}}_{\tau+1} = \mathbf{b} \odot \frac{\mathbf{Az}_\tau}{|\mathbf{Az}_\tau|}, \quad (10.2.1)$$

where \odot is elementwise multiplication, and $\mathbf{b} = |\mathbf{Ax}|$ so that $b_r^2 = y_r$ for all $r = 1, \dots, m$. Then

$$\mathbf{v}_{\tau+1} = \arg \min_{\mathbf{v} \in \mathbb{C}^n} \|\hat{\mathbf{v}}_{\tau+1} - \mathbf{Av}\|. \quad (10.2.2)$$

(In the case of Fourier data, the step (10.2.1)–(10.2.2) essentially adjusts the modulus of the Fourier transform of the current guess so that it fits the measured data.) These two steps can alternatively be written in the form

$$\mathbf{v}_{\tau+1} = \mathcal{P}_A(\mathbf{z}_{\tau+1}) = \mathcal{P}_2(\mathcal{P}_1(\mathbf{z}_{\tau+1})) \quad \text{where} \quad \mathcal{P}_1(\mathbf{z}) = \mathbf{b} \odot \frac{\mathbf{z}}{|\mathbf{z}|}; \mathcal{P}_2(\mathbf{v}) = \mathbf{A}(\mathbf{A}^* \mathbf{A})^{-1} \mathbf{A}^* \mathbf{v}. \quad (10.2.3)$$

Here, the mapping \mathcal{P}_1 corrects the magnitude and \mathcal{P}_2 projects onto the range of \mathbf{A} . Finally, if we know that the solution belongs to a convex set \mathcal{S} (as in the case where the signal is known to be real-valued, possibly non-negative and of finite support), then the next iterate is

$$\mathbf{z}_{\tau+1} = P_{\mathcal{S}}(\mathbf{v}_{\tau+1}), \quad (10.2.4)$$

where $P_{\mathcal{S}}$ is the projection onto the convex set \mathcal{S} . If no such information is available, then $\mathbf{z}_{\tau+1} = \mathbf{v}_{\tau+1}$. The first step (10.2.2) is not a projection onto a convex set and, therefore, it is in general completely unclear whether the Gerchberg-Saxton algorithm actually converges. (And if it were to converge, at what speed?) It is also unclear how the procedure should be initialized to yield accurate final estimates.¹

Now that we have discussed the generalized version of the Gerchberg-Saxton (GS) and Fienup, we shall briefly mention the original form of these algorithms as developed by Gerchberg, Saxton, and Fienup. Gerchberg-Saxton (GS) algorithm [114] and its generalizations are widely popular and have had great impact on different scientific fields [102]. This algorithm was developed for the two-intensity phase retrieval problem where the goal is to recover a signal $\mathbf{x} \in \mathbb{R}^n$ from the intensity of the signal $|\mathbf{x}|$ and its Fourier transform $|\mathcal{F}(\mathbf{x})|$. More precisely, measurements of the form $|\mathbf{x}_k|$ and $|\mathbf{f}_k^* \mathbf{x}|$ for $k = 1, 2, \dots, n$.² The GS algorithm starts from a random estimate of the signal. The updates in (10.2.1) and (10.2.2), specialized to the two-intensity problem take the form

¹Most papers on the topic suggest picking the initial phase at random that is set $\hat{\mathbf{v}}_1 = \mathbf{b} \odot \mathbf{w}$ where each entry of \mathbf{w} is i.i.d. random variable of the form $e^{i\phi}$ with ϕ uniformly random in $[0, 2\pi]$.

²We note that this is an instance of the generalized phase retrieval problem where the sensing matrix is the concatenation of the $n \times n$ identity and DFT matrices.

- (1) take Fourier transform of the estimate of the signal
- (2) replace the modulus of the resulting Fourier transform with the measured Fourier modulus
- (3) inverse Fourier transform the estimate of the Fourier transform
- (4) replace the modulus of the resulting signal with the measured signal modulus

In [103] Fienup developed a generalization of the GS algorithm so as to recover a non-negative signal from the magnitude of its Fourier transform $|\mathcal{F}(\mathbf{x})|$. Fienup's algorithm essentially replaces step (4) of the GS algorithm by enforcing the non-negative constraint on the signal (or other signal domain information which may be available).

10.2.2 Solvent flipping algorithm

The Solvent Flipping (SF) algorithm [4] is obtained by replacing step (10.2.4) of the GS algorithm by $\mathcal{R}_{\mathcal{S}} = 2\mathcal{P}_{\mathcal{S}} - \mathcal{I}$. So that the overall iteration update takes the form

$$\mathbf{z}_{\tau+1} = \mathcal{R}_{\mathcal{S}}(\mathcal{P}_{\mathbf{A}}(\mathbf{z}_{\tau})).$$

10.2.3 Hybrid input-output algorithm

The Hybrid Input-Output (HIO) algorithm [103, 105] uses ideas from nonlinear feedback control theory. The updates in this case are of the form

$$\mathbf{z}_{\tau+1} = \begin{cases} \mathcal{P}_{\mathbf{A}}(\mathbf{z}_{\tau}) & \text{if } \mathbf{z}_{\tau} \in \mathcal{S}, \\ -(\mathcal{I} - \beta\mathcal{P}_{\mathbf{A}})(\mathbf{z}_{\tau}) & \text{otherwise.} \end{cases}$$

We note that HIO was developed to increase the speed of convergence of the ER algorithm. It has been documented to converge faster for both two-intensity and one-intensity measurements. However, [99] indicates that the algorithm may converge to a fixed point \mathbf{z}^3 which may not obey $\mathcal{P}_{\mathcal{S}}(\mathbf{z}) = \mathcal{P}_{\mathbf{A}}(\mathbf{z}) = \mathbf{z}$. That being said, [99] also

³We arrive at a fixed point when $\mathbf{z}_{\tau+1} = \mathbf{z}_{\tau}$.

provides a simple way of fixing this issue. This algorithm is often combined with the ER steps where one performs a few HIO updates and then performs a few iterations of the ER update.

Chapter 11

Phase retrieval via convex relaxation

In this chapter we shall present results that show convex programming techniques are provably effective for the phase retrieval problem in a physically realizable model. To this aim we shall first explain the convex programming approach to phase retrieval in Section 11.1. We continue in Section 11.2 by introducing our physically inspired model and finally state our theoretical results in Section 11.4.

11.1 Convex relaxation

Previous work [52, 69] suggested to bring convex programming techniques to bear on the phase retrieval problem. Returning to the general formulation (8.0.1), the phase problem asks to recover $\mathbf{x} \in \mathbb{C}^n$ subject to data constraints of the form

$$\text{trace}(\mathbf{a}_r \mathbf{a}_r^* \mathbf{x} \mathbf{x}^*) = y_r, \quad r = 1, \dots, m,$$

where $\text{trace}(\mathbf{X})$ is the trace of the matrix \mathbf{X} . The idea is then to lift the problem in higher dimensions: introducing the Hermitian matrix variable $\mathbf{X} \in \mathcal{S}^{n \times n}$, the phase

problem is equivalent to finding \mathbf{X} obeying

$$\mathbf{X} \succeq 0, \quad \text{rank}(\mathbf{X}) = 1, \quad \text{trace}(\mathbf{a}_r \mathbf{a}_r^* \mathbf{X}) = y_r \text{ for } r = 1, \dots, m \quad (11.1.1)$$

where, here and below, $\mathbf{X} \succeq 0$ means that \mathbf{X} is positive semidefinite. This problem is not tractable and, by dropping the rank constraint, is relaxed into

$$\begin{aligned} & \text{minimize} && \text{trace}(\mathbf{X}) \\ & \text{subject to} && \mathbf{X} \succeq 0 \\ & && \text{trace}(\mathbf{a}_r \mathbf{a}_r^* \mathbf{X}) = y_r, \quad r = 1, \dots, m. \end{aligned} \quad (11.1.2)$$

PhaseLift (11.1.2) is a semidefinite program (SDP). If its solution happens to have rank one and is equal to $\mathbf{x}\mathbf{x}^*$, then a simple factorization recovers \mathbf{x} up to a global phase/sign.

We pause to emphasize that in different contexts, similar convex relaxations for optimizing quadratic objectives subject to quadratic constraints are known as Schor's semidefinite relaxations, see [185, Section 4.3] and [115] on the MAXCUT problem from graph theory for spectacular applications of these ideas. For related convex relaxations of quadratic problems, we refer the interested reader to the wonderful tutorial [228].

11.2 Coded diffraction patterns

Imagine then that we modulate the signal before diffraction. Letting $d[t]$ be the modulating waveform, we would observe the diffraction pattern

$$y_r = \left| \sum_{t=0}^{n-1} x[t] \bar{d}[t] e^{-i2\pi\omega_r t} \right|^2, \quad \omega_r \in \Omega. \quad (11.2.1)$$

We call this a *coded diffraction pattern* (CDP) since it gives us information about the spectrum of $\{x[t]\}$ modulated by the code $\{d[t]\}$. There are several ways of achieving modulations of this type: one can use a phase mask just after the sample, see Figure 14.5, or use an optical grating to modulate the illumination beam as mentioned

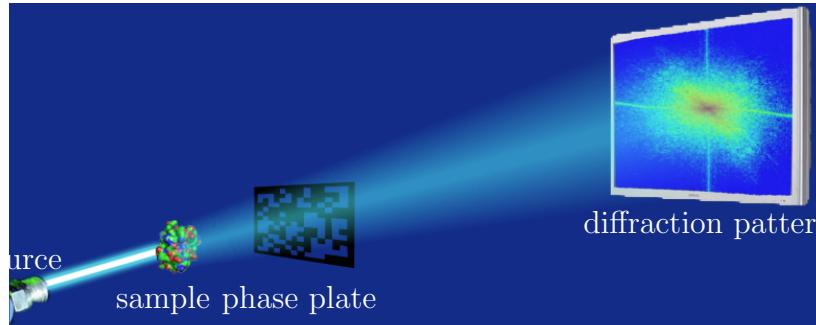


Figure 11.1: An illustrative setup for acquiring coded diffraction patterns.

in [151], or even use techniques from ptychography which scan an illumination patch on an extended specimen [207, 221]. We refer to [52] for a more thorough discussion of such approaches.

In this chapter, we analyze such a data collection scheme in which one uses multiple modulations. Our model for data acquisition is thus as follows:

$$y_r = \left| \sum_{t=0}^{n-1} x[t] \bar{d}_\ell(t) e^{-i2\pi kt/n} \right|^2, \quad r = (\ell, k), \quad \begin{matrix} 0 \leq k \leq n-1 \\ 1 \leq \ell \leq L \end{matrix}; \quad (11.2.2)$$

In words, we collect the magnitude of the discrete Fourier transform (DFT) of L modulations of the signal \mathbf{x} . In matrix notation, letting \mathbf{D}_ℓ be the diagonal matrix with the modulation pattern $d_\ell[t]$ on the diagonal and \mathbf{f}_k^* be the rows of the DFT, we observe

$$y_{\ell,k} = |\mathbf{f}_k^* \mathbf{D}_\ell^* \mathbf{x}|^2.$$

Note that in this case

$$y_r = y_{\ell,k} = |\mathbf{a}_r^* \mathbf{x}|^2,$$

where $\mathbf{a}_r = \mathbf{D}_\ell \mathbf{f}_k$. We prove that if we use random modulation patterns (random waveforms $d[t]$), then the solution to (11.1.2) is exact with high probability provided that we have sufficiently many CDPs. In fact, we will see that the feasible set in

(11.1.2) equal to

$$\{\mathbf{X} : \mathbf{X} \succeq \mathbf{0} \text{ and } \mathcal{A}(\mathbf{X}) = \mathbf{y}\} \quad (11.2.3)$$

reduces to a single point $\mathbf{x}\mathbf{x}^*$. Above $\mathcal{A} : \mathcal{S}^{n \times n} \rightarrow \mathbb{R}^{m=nL}$ ($\mathcal{S}^{n \times n}$ is the space of self-adjoint matrices) is the linear mapping giving us the linear equalities in (11.1.2),

$$\mathcal{A}(\mathbf{X}) = \{\mathbf{f}_k^* \mathbf{D}_\ell^* \mathbf{X} \mathbf{D}_\ell \mathbf{f}_k\}_{\ell,k} = \{\text{trace}(\mathbf{D}_\ell \mathbf{f}_k \mathbf{f}_k^* \mathbf{D}_\ell^* \mathbf{X})\}_{\ell,k}.$$

11.3 Modeling assumptions

Our model assumes random modulations and we work with diagonal matrices \mathbf{D}_ℓ , $1 \leq \ell \leq L$, which are i.i.d. copies of a matrix \mathbf{D} , whose entries are themselves i.i.d. copies of a random variable d . Throughout, we assume that d is symmetric, obeys $|d| \leq M$ as well as the moment conditions

$$\mathbb{E} d = 0, \quad \mathbb{E} d^2 = 0, \quad \mathbb{E} |d|^4 = 2(\mathbb{E} |d|^2)^2. \quad (11.3.1)$$

A random variable obeying these assumptions is said to be *admissible*. The reason why we can have $\mathbb{E} d^2 = 0$ while $d \neq 0$ is that d is complex valued. An example of an admissible random variable is $d = b_1 b_2$, where b_1 and b_2 are independent and distributed as

$$b_1 = \begin{cases} 1 & \text{with prob. } \frac{1}{4} \\ -1 & \text{with prob. } \frac{1}{4} \\ -i & \text{with prob. } \frac{1}{4} \\ i & \text{with prob. } \frac{1}{4} \end{cases} \quad \text{and} \quad b_2 = \begin{cases} \frac{1}{\sqrt{2}} & \text{with prob. } \frac{4}{5} \\ \sqrt{3} & \text{with prob. } \frac{1}{5} \end{cases}. \quad (11.3.2)$$

We would like to emphasize that we impose $\mathbb{E}[d^2] = 0$ mostly to simplify our exposition. In fact, the conclusion of Theorem 11.4.1 below remains valid if $\mathbb{E}[d^2] \neq 0$, although we do not prove this in this paper. In particular, we can also work with d

distributed as

$$d = \begin{cases} \sqrt{2} & \text{with prob. } \frac{1}{4} \\ 0 & \text{with prob. } \frac{1}{2} \\ -\sqrt{2} & \text{with prob. } \frac{1}{4} \end{cases}. \quad (11.3.3)$$

11.4 Main results

In this chapter we work with one dimensional signals. We would like to mention that our methods would extend to higher dimensional signals by using unstructured two dimensional (and higher dimensional) codes, although we do not pursue this in this dissertation. However, one can also see that it is immediate to break down two dimensional (and higher dimensional) phase retrieval problems of the form studied in this paper into one dimensional problems by a “tensorizing” trick which we discuss further in Appendix F. We shall first explain our results in the absence of noise in our measurements and then discuss our results in the presence of noise.

11.4.1 Noiseless measurements

Theorem 11.4.1 *Suppose that the modulation is admissible and that the number L of coded diffraction patterns obeys*

$$L \geq c \cdot \log^4 n,$$

for some fixed numerical constant c . Then with probability at least $1 - 1/n$, the feasibility problem (11.2.3) reduces to a unique point, namely, $\mathbf{x}\mathbf{x}^$, and thus recovers \mathbf{x} up to a global phase shift. For $\gamma \geq 1$, setting $L \geq c\gamma \log^4 n$ leads to a probability of success at least $1 - n^{-\gamma}$.*

Thus, in a stylized physical setting, it is possible to recover an arbitrary signal from a fairly limited number of coded diffraction patterns by solving an SDP feasibility problem.

Mathematically, the phase recovery problem is different than that in which the sampling vectors are Gaussian as in [52]. The reason is that the measurements in Theorem 11.4.1 are far more structured and far ‘less random’. Loosely speaking, our random modulation model uses on the order of $m := nL$ random bits whereas the Gaussian model with the same number of quadratic equations would use on the order of mn random bits (this can be formalized by using the notion of entropy from information theory). A consequence of this difference is that the proof of the theorem requires new techniques and ideas. Having said this, an open and interesting research direction is to close the gap—remove the log factors—and show whether or not perfect recovery can be achieved from a number of coded diffraction patterns independent of dimension.

11.4.2 Noisy measurements

In this section we aim to prove some results concerning the stability of the SDP algorithm vis a vis noise. For this purpose assume that our measurements are corrupted in the sense that there is some noise on each measurements

$$\mathbf{y}_r = |\mathbf{a}_r^* \mathbf{x}|^2 + \mathbf{w}_r,$$

where $\mathbf{w} \in \mathbb{R}^m$ denotes the corruption. In this case we shall use the following natural relaxation that tries to minimize the Euclidean norm of the fit to the measurements

$$\hat{\mathbf{X}} = \arg \min_{\bar{\mathbf{X}}} \sum_{r=1}^m (\mathbf{y}_r - \mathbf{a}_r^* \bar{\mathbf{X}} \mathbf{a}_r)^2 \quad \text{subject to } \bar{\mathbf{X}} \succeq 0. \quad (11.4.1)$$

Theorem 11.4.2 *Suppose that the modulation is admissible and that the number L of coded diffraction patterns obeys*

$$L \geq c \cdot \log^4 n,$$

for some fixed numerical constant c . Then with probability at least $1 - 1/n$, the optimal

solution of problem (11.4.1) obeys

$$\|\hat{\mathbf{X}} - \mathbf{x}\mathbf{x}^*\| \leq C \|\mathbf{w}\|_{\ell_2},$$

where C is a fixed numerical constant. For $\gamma \geq 1$, setting $L \geq c\gamma \log^4 n$ leads to a probability of success at least $1 - n^{-\gamma}$.

The above result shows that the convex programming approach to phase retrieval is robust vis a vis noise for a physically realizable model. We note that this is the first result that establishes stable phase retrieval with a physically realizable model for any algorithm. However, we note that there is room for improvement in this result as the author conjectures that the optimal results should be $\|\hat{\mathbf{X}} - \mathbf{x}\mathbf{x}^*\| \leq C \|\mathbf{w}\|_{\ell_2} / \sqrt{m}$. In the author's view establishing this conjecture is a very interesting open problem.¹ We note that [55] establishes this conjecture with Gaussian measurements. However, in the author's view establishing the sharp noise bound for the CDP model is significantly more challenging.

11.5 Comparison with previous work

In this section we shall briefly comment on other results that use similar convex relaxations and then briefly compare our result with other related works.

11.5.1 Comparison with prior art using convex relaxation

Starting with [64], a line of work established that if the sampling vectors \mathbf{a}_k are sufficiently randomized, then the convex relaxation is provably exact. Assuming that the \mathbf{a}_r 's are independent random (complex-valued) Gaussian vectors, [64] shows that on the order of $n \log n$ quadratic measurements are sufficient to guarantee perfect recovery via (11.1.2) with high probability. In subsequent work, [55] reached the same conclusion from on the order of n equations only, by solving the SDP feasibility

¹The author strongly believes that the mathematical tools required to establish this result will have implications for many other problems as well.

problem; to be sure, [55] establishes that the set of matrices obeying the constraints in (11.1.2) reduces to a unique point namely, $\mathbf{x}\mathbf{x}^*$, see [82] for a similar result.² Finally, inspired by PhaseLift and the famous MAXCUT relaxation of Goemans and Williamson, [237] proposed another semidefinite relaxation called PhaseCut whose performance from noiseless data—in terms of the number of samples needed to achieve perfect recovery—turns out to be identical to that of PhaseLift.

While this is all reassuring, the problem is that the Gaussian model, in which each measurement gives us the magnitude of the dot product $\sum_{t=0}^{n-1} x[t]a_k[t]$ between the signal and (complex-valued) Gaussian white noise, is very far from the kind of data one can collect in an X -ray imaging and many related experiments. The purpose of this chapter was to show that convex relaxation is still exact in a physically inspired setup where one can modulate the signal of interest and then let diffraction occur.

The first version of the results of this chapter was made publicly available at the same time as [120], which begins to study the performance of PhaseLift from non-Gaussian sampling vectors. There, the authors study sampling schemes from certain finite vector configurations, dubbed t-designs. These models are different from ours and do not include our coded diffraction patterns as a special case. Hence, our results are not comparable.

11.5.2 Other approaches to phase retrieval and related works

There are of course other approaches to phase retrieval and we mention some literature for completeness and to inform the interested reader of recent progress in this area. Balan [18] studies a problem where the sampling vectors model a short-time Fourier transform. Balan, Casazza and Edidin [22] formulate the phase retrieval problem as nonconvex optimization problem. In [21], the same authors [21] describe some applications of the phase problem in signal processing and speech analysis and presents some necessary and sufficient conditions which guarantee that the solution to (8.0.1) is unique. As we discuss in Section 10.1 other articles studying the minimal number of frame coefficient magnitudes for noiseless recovery include [11, 19, 26, 50, 179].

² [55] also establishes near-optimal estimation bounds from noisy data as mentioned earlier.

We recommend the two blog posts [2] and [3] by Mixon and the references therein for a comprehensive review and discussion of such results. Lower bounds on the performance of any recovery method from noisy data are studied in [23, 26, 94].

On the algorithmic side, [17] proposes a nonlinear scheme for phase retrieval having exponential time complexity in the dimension of the signal \mathbf{x} while [20] presents a tractable algorithm requiring a number of measurements at least quadratic in the dimension of the signal; that is to say, $m \geq c \cdot n^2$ for some constant $c > 0$.

There also is a recent body of work studying the phase retrieval under sparsity assumptions about the signal we wish to recover, see [134, 148, 192, 195, 216] as well as the references therein. Finally, a different line of work [11, 27] studies the phase retrieval by polarization, see also [204] for a related approach. This technique comes with an algorithm that can achieve recovery using on the order of $\log n$ specially constructed masks/codes in the noiseless case. However, to the extent of our knowledge, PhaseLift offers more flexibility in terms of the number and types of masks that can be used since it can be applied regardless of the data acquisition scheme. In addition, when dealing with noisy data PhaseLift behaves very well, see Section 14.2.3 below and the experiments in [27].

11.6 Discussion

In this chapter, we proved that a signal could be recovered by convex programming techniques from a few diffraction patterns corresponding to generic modulations obeying an admissibility condition. We expect that our results, methods and proofs extend to more general random modulations although we have not pursued such extensions in this paper. Further, we proved that on the order of $(\log n)^4$ CDPs suffice for perfect recovery and we expect that further refinements would allow to reduce this number, perhaps all the way down to a figure independent of the number n of unknowns. Such refinements appear quite involved to us and since our intention is to provide a reasonably short and conceptually simple argument, we leave such refinements to

future research.³

³We note that after the first publication of our results in [58], [121] showed that by simple yet insightful modifications to our arguments the number of required patterns can be furthered reduced to the order of $(\log n)^2$ patterns.

Chapter 12

Phase retrieval via Wirtinger Flow

In the previous chapter we discussed one possible approach to phase retrieval based on convex programming via SDPs. While in principle SDP based relaxations offer tractable solutions, they become computationally prohibitive as the dimension of the signal increases. Indeed, for a large number of unknowns the memory requirements are far out of reach of almost all computers so that these SDP relaxations are de facto impractical. To overcome these challenges in Section 12.1 we present an alternative solution to the phase retrieval problem based on non-convex optimization which we call Wirtinger Flow. We then establish in Section 12.2 that even though this algorithm is non-convex it still converges to the global optimum at a geometric rate. Furthermore, in Section 12.3 we show that this algorithm is also stable vis a vis noise. Finally, we end this chapter by comparing our approach to phase retrieval with some other non-convex schemes in Section 12.4. We shall see in the next chapter how

12.1 Algorithm: Wirtinger Flow

This chapter introduces an approach to phase retrieval based on non-convex optimization as well as a solution algorithm, which has two components: (1) a careful initialization obtained by means of a spectral method, and (2) a series of updates refining this initial estimate by iteratively applying a novel update rule, much like in

a gradient descent scheme. We refer to the combination of these two steps, introduced in reverse order below, as the *Wirtinger Flow* (WF) algorithm.

12.1.1 Minimization of a non-convex objective

Let $\ell(x, y)$ be a loss function measuring the misfit between both its scalar arguments. If the loss function is non-negative and vanishes only when $x = y$, then a solution to the generalized phase retrieval problem (8.0.2) is any solution to

$$\text{minimize } f(\mathbf{z}) := \frac{1}{2m} \sum_{r=1}^m \ell(y_r, |\mathbf{a}_r^* \mathbf{z}|^2), \quad \mathbf{z} \in \mathbb{C}^n. \quad (12.1.1)$$

Although one could study many loss functions, we shall focus in this paper on the simple quadratic loss $\ell(x, y) = (x - y)^2$. Admittedly, the formulation (12.1.1) does not make the problem any easier since the function f is not convex. Minimizing non-convex objectives, which may have very many stationary points, is known to be NP-hard in general. In fact, even establishing convergence to a local minimum or stationary point can be quite challenging, please see [182] for an example where convergence to a local minimum of a degree-four polynomial is known to be NP-hard.¹ As a side remark, deciding whether a stationary point of a polynomial of degree four is a local minimizer is already known to be NP-hard.

Our approach to (12.1.1) is simply stated: start with an initialization \mathbf{z}_0 , and for $\tau = 0, 1, 2, \dots$, inductively define

$$\mathbf{z}_{\tau+1} = \mathbf{z}_\tau - \frac{\mu_{\tau+1}}{\|\mathbf{z}_0\|_{\ell_2}^2} \left(\frac{1}{m} \sum_{r=1}^m (|\mathbf{a}_r^* \mathbf{z}|^2 - y_r) (\mathbf{a}_r \mathbf{a}_r^*) \mathbf{z} \right) := \mathbf{z}_\tau - \frac{\mu_{\tau+1}}{\|\mathbf{z}_0\|_{\ell_2}^2} \nabla f(\mathbf{z}_\tau). \quad (12.1.2)$$

If the decision variable \mathbf{z} and the sampling vectors were all real valued, the term between parentheses would be the gradient of f , as our notation suggests. However, since $f(\mathbf{z})$ is a mapping from \mathbb{C}^n to \mathbb{R} , it is not holomorphic and hence not complex-differentiable. However, this term can still be viewed as a gradient based on Wirtinger derivatives reviewed in Appendix G. Hence, (12.1.2) is a form of steepest descent and

¹Observe that if all the sampling vectors are real valued, our objective is also a degree-four polynomial.

the parameter $\mu_{\tau+1}$ can be interpreted as a step size (note nonetheless that the effective step size is also inversely proportional to the magnitude of the initial guess).

12.1.2 Initialization via a spectral method

Our main result states that for a certain random model, if the initialization \mathbf{z}_0 is sufficiently accurate, then the sequence $\{\mathbf{z}_\tau\}$ will converge toward a solution to the generalized phase problem (8.0.2). In this chapter, we propose computing the initial guess \mathbf{z}_0 via a spectral method, detailed in Algorithm 7. In words, \mathbf{z}_0 is the leading eigenvector of the positive semidefinite Hermitian matrix $\sum_r y_r \mathbf{a}_r \mathbf{a}_r^*$ constructed from the knowledge of the sampling vectors and observations. (As usual, \mathbf{a}_r^* is the adjoint of \mathbf{a}_r .) Letting \mathbf{A} be the $m \times n$ matrix whose r th row is \mathbf{a}_r^* so that with obvious notation $\mathbf{y} = |\mathbf{A}\mathbf{x}|^2$, \mathbf{z}_0 is the leading eigenvector of $\mathbf{A}^* \text{diag}\{\mathbf{y}\} \mathbf{A}$ and can be computed via the power method by repeatedly applying \mathbf{A} , entrywise multiplication by \mathbf{y} and \mathbf{A}^* . In the theoretical framework we study below, a constant number of power iterations would give machine accuracy because of an eigenvalue gap between the top two eigenvalues, please see Appendix I for additional information.

Algorithm 7 Wirtinger Flow: Initialization

Input: Observations $\{y_r\} \in \mathbb{R}^m$.

Set

$$\lambda^2 = n \frac{\sum_r y_r}{\sum_r \|\mathbf{a}_r\|_{\ell_2}^2}.$$

Set \mathbf{z}_0 , normalized to $\|\mathbf{z}_0\| = \lambda$, to be the eigenvector corresponding to the largest eigenvalue of

$$\mathbf{Y} = \frac{1}{m} \sum_{r=1}^m y_r \mathbf{a}_r \mathbf{a}_r^*.$$

Output: Initial guess \mathbf{z}_0 .

12.1.3 Wirtinger flow as a stochastic gradient scheme

We would like to motivate the Wirtinger flow algorithm and provide some insight as to why we expect it to work in a model where the sampling vectors are random. First, we emphasize that our statements in this section are heuristic in nature; as it will become clear in the proof Section 15.3, a correct mathematical formalization of these ideas is far more complicated than our heuristic development here may suggest. Second, although our ideas are broadly applicable, it makes sense to begin understanding the algorithm in a setting where everything is real valued, and in which the vectors a_r are i.i.d. $\mathcal{N}(\mathbf{0}, \mathbf{I})$.

Let \mathbf{x} be a solution to (8.0.2) so that $y_r = |\langle \mathbf{a}_r, \mathbf{x} \rangle|^2$, and consider the initialization step first. In the Gaussian model, a simple moment calculation gives

$$\mathbb{E} \left[\frac{1}{m} \sum_{r=1}^m y_r \mathbf{a}_r \mathbf{a}_r^* \right] = \mathbf{I} + 2\mathbf{x}\mathbf{x}^*.$$

By the strong law of large numbers, the matrix \mathbf{Y} in Algorithm 7 is equal to the right-hand side in the limit of large samples. Since any leading eigenvector of $\mathbf{I} + 2\mathbf{x}\mathbf{x}^*$ is of the form $\lambda\mathbf{x}$ for some scalar $\lambda \in \mathbb{R}$, we see that the initialization step would recover \mathbf{x} perfectly, up to a global sign or phase factor, had we infinitely many samples. Indeed, the chosen normalization would guarantee that the recovered signal is of the form $\pm\mathbf{x}$. As an aside, we would like to note that the top two eigenvalues of $\mathbf{I} + 2\mathbf{x}\mathbf{x}^*$ are well separated unless $\|\mathbf{x}\|$ is very small, and that their ratio is equal to $1+2\|\mathbf{x}\|^2$. Now with a finite amount of data, the leading eigenvector of \mathbf{Y} will of course not be perfectly correlated with \mathbf{x} but we hope that it is sufficiently correlated to point us in the right direction.

We now turn our attention to the gradient-update (12.1.2) and define

$$F(\mathbf{z}) = \frac{1}{2} \mathbf{z}^* (\mathbf{I} - \mathbf{x}\mathbf{x}^*) \mathbf{z} + (\|\mathbf{z}\|_{\ell_2}^2 - \|\mathbf{x}\|_{\ell_2}^2)^2,$$

where here and below, \mathbf{x} is once again our planted solution. The first term ensures that the direction of \mathbf{z} matches the direction of \mathbf{x} and the second term penalizes the deviation of the Euclidean norm of \mathbf{z} from that of \mathbf{x} . Obviously, the minimizers of

this function are $\pm \mathbf{x}$. Now consider the gradient scheme

$$\mathbf{z}_{\tau+1} = \mathbf{z}_\tau - \frac{\mu_{\tau+1}}{\|\mathbf{z}_0\|_{\ell_2}^2} \nabla F(\mathbf{z}_\tau). \quad (12.1.3)$$

In Section 15.3.9, we show that if $\min \|\mathbf{z}_0 \pm \mathbf{x}\|_{\ell_2} \leq 1/8 \|\mathbf{x}\|_{\ell_2}$, then $\{\mathbf{z}_\tau\}$ converges to \mathbf{x} up to a global sign. However, this is all ideal as we would need knowledge of \mathbf{x} itself to compute the gradient of F ; we simply cannot run this algorithm.

Consider now the WF update and assume for a moment that \mathbf{z}_τ is fixed and independent of the sampling vectors. We are well aware that this is a false assumption but nevertheless wish to explore some of its consequences. In the Gaussian model, if \mathbf{z} is independent of the sampling vectors, then Lemma 15.3.2 shows that $\mathbb{E}[\nabla f(\mathbf{z})] = \nabla F(\mathbf{z})$ and, therefore,

$$\mathbb{E}[\mathbf{z}_{\tau+1}] = \mathbb{E}[\mathbf{z}_\tau] - \frac{\mu_{\tau+1}}{\|\mathbf{z}_0\|_{\ell_2}^2} \mathbb{E}[\nabla f(\mathbf{z}_\tau)] \quad \Rightarrow \quad \mathbb{E}[\mathbf{z}_{\tau+1}] = \mathbf{z}_\tau - \frac{\mu_{\tau+1}}{\|\mathbf{z}_0\|_{\ell_2}^2} \nabla F(\mathbf{z}_\tau).$$

Hence, the average WF update is the same as that in (12.1.3) so that we can interpret the Wirtinger flow algorithm as a stochastic gradient scheme in which we only get to observe an unbiased estimate $\nabla f(\mathbf{z})$ of the “true” gradient $\nabla F(\mathbf{z})$.

Regarding WF as a stochastic gradient scheme helps us in choosing the learning parameter or step size μ_τ . Lemma 15.3.7 asserts that

$$\|\nabla f(\mathbf{z}) - \nabla F(\mathbf{z})\|_{\ell_2}^2 \leq \|\mathbf{x}\|_{\ell_2}^2 \cdot \min \|\mathbf{z} \pm \mathbf{x}\|_{\ell_2} \quad (12.1.4)$$

holds with high probability. Looking at the right-hand side, this says that the uncertainty about the gradient estimate depends on how far we are from the actual solution \mathbf{x} . The further away, the larger the uncertainty or the noisier the estimate. This suggests that in the early iterations we should use a small learning parameter as the noise is large since we are not yet close to the solution. However, as the iterations count increases and we make progress, the size of the noise also decreases and we can pick larger values for the learning parameter. This heuristic together with

experimentation lead us to consider

$$\mu_\tau = \min(1 - e^{-\tau/\tau_0}, \mu_{\max}) \quad (12.1.5)$$

shown in Figure 12.1. Values of τ_0 around 330 and of μ_{\max} around 0.4 worked well in our simulations. This makes sure that μ_τ is rather small at the beginning (e.g. $\mu_1 \approx 0.003$ but quickly increases and reaches a maximum value of about 0.4 after 200 iterations or so.

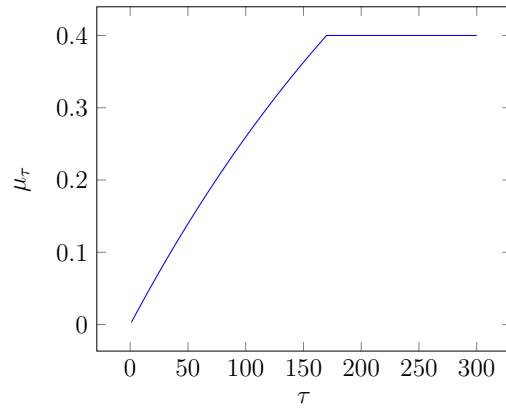


Figure 12.1: Learning parameter μ_τ from (12.1.5) as a function of the iteration count τ ; here, $\tau_0 \approx 330$ and $\mu_{\max} = 0.4$.

12.2 Exact phase retrieval via Wirtinger flow

In this section we study the performance of the WF algorithm when there is no noise in our model. Our main result in Section 12.2.1 establishes the correctness of the Wirtinger flow algorithm in the Gaussian model (also defined in Section 12.2.1). Later in Section 12.2.2, we shall also develop exact recovery results for the CDP model.

We need to define the distance to the solution set.

Definition 12.2.1 Let $\mathbf{x} \in \mathbb{C}^n$ be any solution to the quadratic system (8.0.2) (the

signal we wish to recover). For each $\mathbf{z} \in \mathbb{C}^n$, define

$$\text{dist}(\mathbf{z}, \mathbf{x}) = \min_{\phi \in [0, 2\pi]} \|\mathbf{z} - e^{i\phi} \mathbf{x}\|_{\ell_2}.$$

12.2.1 Theory for the Gaussian model

Our main result shows the correctness of the Wirtinger flow algorithm in the Gaussian model defined below.

Definition 12.2.2 We say that the sampling vectors follow the Gaussian model if $\mathbf{a}_r \in \mathbb{C}^n \stackrel{\text{i.i.d.}}{\sim} \mathcal{N}(\mathbf{0}, \mathbf{I}/2) + i\mathcal{N}(\mathbf{0}, \mathbf{I}/2)$. In the real-valued case, they are i.i.d. $\mathcal{N}(\mathbf{0}, \mathbf{I})$.

Theorem 12.2.3 Let \mathbf{x} be an arbitrary vector in \mathbb{C}^n and $\mathbf{y} = |\mathbf{A}\mathbf{x}|^2 \in \mathbb{R}^m$ be m quadratic samples with $m \geq c_0 \cdot n \log n$, where c_0 is a sufficiently large numerical constant. Then the Wirtinger flow initial estimate \mathbf{z}_0 normalized to have squared Euclidean norm equal to $m^{-1} \sum_r y_r$,² obeys

$$\text{dist}(\mathbf{z}_0, \mathbf{x}) \leq \frac{1}{8} \|\mathbf{x}\|_{\ell_2} \tag{12.2.1}$$

with probability at least $1 - 10e^{-\gamma n} - 8/n^2$ (γ is a fixed positive numerical constant). Further, take a constant learning parameter sequence, $\mu_\tau = \mu$ for all $\tau = 1, 2, \dots$ and assume $\mu \leq c_1/n$ for some fixed numerical constant c_1 . Then there is an event of probability at least $1 - 13e^{-\gamma n} - me^{-1.5m} - 8/n^2$, such that on this event, starting from any initial solution \mathbf{z}_0 obeying (12.2.1), we have

$$\text{dist}(\mathbf{z}_\tau, \mathbf{x}) \leq \frac{1}{8} \left(1 - \frac{\mu}{4}\right)^{\tau/2} \cdot \|\mathbf{x}\|_{\ell_2}.$$

Clearly, one would need $2n$ quadratic measurements to have any hope of recovering $\mathbf{x} \in \mathbb{C}^n$. It is also known that in our sampling model, the mapping $\mathbf{z} \mapsto |\mathbf{A}\mathbf{z}|^2$ is injective for $m \geq 4n$ [21] and that this property holds for generic sampling vectors [77].³

²The same results holds with the intialization from Algorithm 7 because $\sum_r \|a_r\|^2 \approx m \cdot n$ with a standard deviation of about the square root of this quantity.

³It is not within the scope of this paper to explain the meaning of generic vectors and, instead, refer the interested reader to [77].

Hence, the Wirtinger flow algorithm loses at most a logarithmic factor in the *sampling complexity*. In comparison, the SDP relaxation only needs a sampling complexity proportional to n (no logarithmic factor) [55], and it is an open question whether Theorem 12.2.3 holds in this regime.

Setting $\mu = c_1/n$ yields ϵ accuracy in a relative sense, namely, $\text{dist}(\mathbf{z}, \mathbf{x}) \leq \epsilon \|\mathbf{x}\|_{\ell_2}$, in $\mathcal{O}(n \log 1/\epsilon)$ iterations. The computational work at each iteration is dominated by two matrix-vector products of the form $\mathbf{A}\mathbf{z}$ and $\mathbf{A}^*\mathbf{v}$, see Appendix I. It follows that the overall computational complexity of the WF algorithm is $\mathcal{O}(mn^2 \log 1/\epsilon)$. In the next section, we will exhibit a modification to the WF algorithm of mere theoretical interest, which also yields exact recovery under the same sampling complexity and an $\mathcal{O}(mn \log 1/\epsilon)$ computational complexity; that is to say, the computational workload is now just *linear* in the problem size.

12.2.2 Theory for the Coded Diffraction Model

We complement our study with theoretical results applying to the model of coded diffraction patterns (Please see Section 11.2 for a detailed description of this model). These results concern a variation of the Wirtinger flow algorithm: whereas the iterations are exactly the same as (12.1.2), the initialization applies an iterative scheme which uses fresh sets of sample at each iteration. This is described in Algorithm 8. In the CDP model, the partitioning assigns to the same group all the observations and sampling vectors corresponding to a given realization of the random code. This is equivalent to partitioning the random patterns into $B + 1$ groups. As a result, sampling vectors in distinct groups are stochastically independent.

Theorem 12.2.4 *Let \mathbf{x} be an arbitrary vector in \mathbb{C}^n and assume we collect L admissible coded diffraction patterns with $L \geq c_0 \cdot (\log n)^4$, where c_0 is a sufficiently large numerical constant. Then the initial solution \mathbf{z}_0 of Algorithm 8 obeys*

$$\text{dist}(\mathbf{z}_0, \mathbf{x}) \leq \frac{1}{8\sqrt{n}} \|\mathbf{x}\|_{\ell_2} \quad (12.2.2)$$

with probability at least $1 - (4L + 2)/n^3$. Further, take a constant learning parameter

Algorithm 8 Initialization via resampled Wirtinger Flow

Input: Observations $\{y_r\} \in \mathbb{R}^m$ and number of blocks B .

Partition the observations and sampling vectors $\{y_r\}_{r=1}^m$ and $\{\mathbf{a}_r\}_{r=1}^m$ into $B + 1$ groups of size $m' = \lfloor m/(B + 1) \rfloor$. For each group $b = 0, 1, \dots, B$, set

$$f(\mathbf{z}; b) = \frac{1}{2m'} \sum_{r=1}^{m'} \left(y_r^{(b)} - |\langle \mathbf{a}_r^{(b)}, \mathbf{z} \rangle|^2 \right)^2,$$

where $\{\mathbf{a}_r^{(b)}\}$ are those sampling vectors belonging to the b th group (and likewise for $\{y_r^{(b)}\}$).

Initialize $\tilde{\mathbf{u}}_0$ to be eigenvector corresponding to the largest eigenvalue of

$$\mathbf{Y} = \frac{1}{m'} \sum_{r=1}^{m'} y_r^{(0)} \mathbf{a}_r^{(0)} \mathbf{a}_r^{(0)*}$$

normalized as in Algorithm (7).

Loop:

for $b = 0$ **to** $B - 1$ **do**

$$\mathbf{u}_{b+1} = \mathbf{u}_b - \frac{\tilde{\mu}}{\|\mathbf{u}_0\|_{\ell_2}^2} \nabla f(\mathbf{u}_b; b)$$

end for

Output: $\mathbf{z}_0 = \mathbf{u}_B$.

sequence, $\mu_\tau = \mu$ for all $\tau = 1, 2, \dots$ and assume $\mu \leq c_1$ for some fixed numerical constant c_1 . Then there is an event of probability at least $1 - (2L + 1)/n^3 - 1/n^2$, such that on this event, starting from any initial solution \mathbf{z}_0 obeying (12.2.2), we have

$$\text{dist}(\mathbf{z}_\tau, \mathbf{x}) \leq \frac{1}{8\sqrt{n}} \left(1 - \frac{\mu}{3} \right)^{\tau/2} \cdot \|\mathbf{x}\|_{\ell_2}. \quad (12.2.3)$$

In the Gaussian model, both statements also hold with high probability provided that $m \geq c_0 \cdot n (\log n)^2$, where c_0 is a sufficiently large numerical constant.

Hence, we achieve perfect recovery from on the order of $n(\log n)^4$ samples arising from a coded diffraction experiment. In Theorem 11.4.1 we established that

PhaseLift—the SDP relaxation—is also exact with a sampling complexity on the order of $n(\log n)^4$ (this has recently been improved to $n(\log n)^2$ [121]). We believe that the sampling complexity of both approaches (WF and SDP) can be further reduced to $n \log n$ (or even n for certain kind of patterns). We leave this to future research.

Setting $\mu = c_1$ yields ϵ accuracy in $\mathcal{O}(\log 1/\epsilon)$ iterations. As the computational work at each iteration is dominated by two matrix-vector products of the form $\mathbf{A}\mathbf{z}$ and $\mathbf{A}^*\mathbf{v}$, it follows that the overall computational is at most $\mathcal{O}(nL \log n \log 1/\epsilon)$. In particular, this approach yields a near-linear time algorithm in the CDP model (linear in the dimension of the signal n). In the Gaussian model, the complexity scales like $\mathcal{O}(mn \log 1/\epsilon)$.

12.3 Stable phase retrieval via Wirtinger flow

In this section we aim to prove some results concerning the stability of Wirtinger flow in the Gaussian model. For this purpose assume that our measurements are corrupted in the sense that there is some noise on each measurements

$$\mathbf{y}_r = |\mathbf{a}_r^* \mathbf{x}|^2 + \mathbf{w}_r,$$

where $\mathbf{w} \in \mathbb{R}^m$ denotes the corruption. In this case we shall use the same non-convex objective of the Wirtinger Flow algorithm (unchanged from the noiseless case)

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{z} \in \mathbb{C}^n} f(\mathbf{z}) := \frac{1}{2m} \sum_{r=1}^m (y_r - |\mathbf{a}_r^* \mathbf{z}|^2)^2. \quad (12.3.1)$$

Our first result establishes that this procedure is accurate.

Theorem 12.3.1 *Let \mathbf{x} be an arbitrary vector in \mathbb{R}^n and $\mathbf{y} = |\mathbf{A}\mathbf{x}|^2 + \mathbf{w} \in \mathbb{R}^m$ be m quadratic samples with $m \geq c_0 \cdot n$, where c_0 is a sufficiently large numerical constant. Then for all $\mathbf{x} \in \mathbb{C}^n$, the solution to (12.3.1) obeys*

$$dist(\hat{\mathbf{x}}, \mathbf{x}) \leq 4 \frac{1}{\sqrt{m}} \frac{\|\mathbf{w}\|_{\ell_2}}{\|\mathbf{x}\|_{\ell_2}},$$

with probability at least $1 - e^{-\gamma_0 m}$ with γ_0 a fixed numerical constant.

We note that this result is optimal (up to unknown constants) and can not possibly be improved. We note that [55] established a similar result for stable versions of PhaseLift.

While the above result is optimal it is not very practical in the sense that it is not clear how to get to the global optimal of (12.3.1). Next we show that the WF algorithm converges to this global optimum at a geometric rate. However, for this result we make two additional assumptions:

- *Real values.* We assume both the measurement vectors \mathbf{a}_r and the unknown signal \mathbf{x} are real valued and in \mathbb{R}^n .
- *Corruption with Gaussian noise.* We also assume that the corruption vector $\mathbf{w} \in \mathbb{R}^m$ is distributed as $\mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$.

We note that the above two restrictions are not really necessary and stronger results are possible. However, the author decided to state and prove the theorem below under these simpler assumptions due to his time constraints. In a yet unpublished note the author removes/weakens these assumptions with significantly more sophisticated arguments.

Theorem 12.3.2 *Let \mathbf{x} be an arbitrary vector in \mathbb{C}^n and $\mathbf{y} = |\mathbf{A}\mathbf{x}|^2 + \mathbf{w} \in \mathbb{R}^m$ be m quadratic samples with $m \geq c_0 \cdot n \log n$, where c_0 is a sufficiently large numerical constant. Also assume that the two assumptions stated above hold. Then the Wirtinger flow initial estimate \mathbf{z}_0 normalized to have squared Euclidean norm equal to $m^{-1} \sum_r y_r$,⁴ obeys*

$$\text{dist}(\mathbf{z}_0, \hat{\mathbf{x}}) \leq \frac{1}{40} \|\mathbf{x}\|_{\ell_2} + c_1 \sigma \quad (12.3.2)$$

with probability at least $1 - 5ne^{-\gamma n} - 9/n^2$ (γ and c_1 are fixed positive numerical constants). Further, take a constant learning parameter sequence, $\mu_\tau = \mu$ for all

⁴The same results holds with the initialization from Algorithm 7 because $\sum_r \|a_r\|^2 \approx m \cdot n$ with a standard deviation of about the square root of this quantity.

$\tau = 1, 2, \dots$ and assume $\mu \leq c_1/n$ for some fixed numerical constant c_1 . Furthermore, assume that

$$\|\mathbf{x}\|_{\ell_2} \geq c_2 \sigma. \quad (12.3.3)$$

holds for a sufficiently large numerical constant c_2 . Then there is an event of probability at least $1 - 13e^{-\gamma n} - me^{-1.5m} - 8/n^2$, such that on this event, starting from any initial solution \mathbf{z}_0 obeying (12.2.1), we have

$$\text{dist}(\mathbf{z}_\tau, \hat{\mathbf{x}}) \leq \frac{1}{20} (1 - 1.5\mu)^{\tau/2} \cdot \|\mathbf{x}\|_{\ell_2}. \quad (12.3.4)$$

First, we would like to point out that the condition (12.3.3) is not overly restrictive as it essentially states that the signal to noise ratio (SNR) needs to be bounded below by a fixed numerical constant. To see how equation (12.3.3) is related to SNR note that $\sum_{r=1}^m |\mathbf{a}_r^* \mathbf{x}|^2 \approx m \|\mathbf{x}\|_{\ell_2}^2$. Therefore, the condition in (12.3.3) is equivalent to

$$\frac{\sum_{r=1}^m |\mathbf{a}_r^* \mathbf{x}|^2}{\|\mathbf{w}\|_{\ell_2}^2} \approx \frac{m \|\mathbf{x}\|_{\ell_2}^2}{m\sigma^2} \geq c_1.$$

Second, (12.3.4) establishes geometric convergence to the global optimum of (12.3.1). Therefore, combining the theorem above with Theorem 12.3.1 we can conclude that

$$\text{dist}(\mathbf{z}_\tau, \mathbf{x}) \leq \frac{1}{20} (1 - 1.5\mu)^{\tau/2} \cdot \|\mathbf{x}\|_{\ell_2} + 4 \frac{1}{\sqrt{m}} \frac{\|\mathbf{w}\|_{\ell_2}}{\|\mathbf{x}\|_{\ell_2}},$$

for a fixed numerical constant C . Therefore, the two theorems above collectively show that the WF scheme is also stable to corruption. The reason this is particularly interesting is that WF is non-convex and it is a common misconception that non-convex algorithms are not stable. Indeed, this incorrect perception together with the fact that convergence to the global optimal of non-convex optimization problems are not well understood, is one of the main reasons that the main focus of the optimization literature has been on convex programming. The above results debunks these incorrect perceptions in the particular case of phase retrieval. Indeed, numerical experiments

in Section 14.2.3 suggest that WF is significantly more stable than its convex counter parts.

12.4 Comparison with other non-convex schemes

We now pause to comment on a few other non-convex schemes in the literature. Other comparisons may be found in Section 11.5.

Earlier, we discussed the error reduction algorithm. The first step (10.2.2) is not a projection onto a convex set and, therefore, it is in general completely unclear whether the ER algorithm actually converges. (And if it were to converge, at what speed?) It is also unclear how the procedure should be initialized to yield accurate final estimates. This is in contrast to the Wirtinger flow algorithm, which in the Gaussian model is shown to exhibit geometric convergence to the solution to the phase retrieval problem. Another benefit is that the Wirtinger flow algorithm does not require solving a least-squares problem (10.2.2) at each iteration; each step enjoys a reduced computational complexity.

A recent contribution related to ours is the interesting paper [187], which proposes an alternating minimization scheme named AltMinPhase for the general phase retrieval problem. AltMinPhase is inspired by the ER update (10.2.1)–(10.2.2) as well as other established alternating projection heuristics [105, 161, 162, 169, 173, 249]. We describe the algorithm in the setup of Theorem 12.2.3 for which [187] gives theoretical guarantees. To begin with, AltMinPhase partitions the sampling vectors \mathbf{a}_r (the rows of the matrix \mathbf{A}) and corresponding observations y_r into $B + 1$ disjoint blocks $(\mathbf{y}^{(0)}, \mathbf{A}^{(0)}), (\mathbf{y}^{(1)}, \mathbf{A}^{(1)}), \dots, (\mathbf{y}^{(B)}, \mathbf{A}^{(B)})$ of roughly equal size. Hence, distinct blocks are stochastically independent from each other. The first block $(\mathbf{y}^{(0)}, \mathbf{A}^{(0)})$ is used to compute an initial estimate \mathbf{z}_0 . After initialization, AltMinPhase goes through a series of iterations of the form (10.2.1)–(10.2.2), however, with the key difference that each iteration uses a fresh set of sampling vectors and observations: in details,

$$\mathbf{z}_{\tau+1} = \arg \min_{\mathbf{z} \in \mathbb{C}^n} \|\hat{\mathbf{v}}_{\tau+1} - \mathbf{A}^{(\tau+1)} \mathbf{z}\|, \quad \hat{\mathbf{v}}_{\tau+1} = \mathbf{b} \odot \frac{\mathbf{A}^{(\tau+1)} \mathbf{z}_\tau}{|\mathbf{A}^{(\tau+1)} \mathbf{z}_\tau|}. \quad (12.4.1)$$

As for the ER algorithm, each iteration requires solving a least-squares problem. Now assume a real-valued Gaussian model as well as a real valued solution $x \in \mathbb{R}^n$. The main result in [187] states that if the first block $(\mathbf{y}^{(0)}, \mathbf{A}^{(0)})$ contains at least $c \cdot n \log^3 n$ samples and each consecutive block contains at least $c \cdot n \log n$ samples— c here denotes a positive numerical constant whose value may change at each occurrence—then it is possible to initialize the algorithm via data from the first block in such a way that each consecutive iterate (12.4.1) decreases the error $\|\mathbf{z}_\tau - \mathbf{x}\|$ by 50%; naturally, all of this holds in a probabilistic sense. Hence, one can get ϵ accuracy in the sense introduced earlier from a total of $c \cdot n \log n \cdot (\log^2 n + \log 1/\epsilon)$ samples. Whereas the Wirtinger flow algorithm achieves arbitrary accuracy from just $c \cdot n \log n$ samples, these theoretical results would require an infinite number of samples. This is, however, not the main point.

The main point is that in practice, it is not realistic to imagine (1) that we will divide the samples in distinct blocks (how many blocks should we form a priori? of which sizes?) and (2) that we will use measured data only once. With respect to the latter, observe that the ER procedure (10.2.1)–(10.2.2) uses all the samples at each iteration. This is the reason why AltMinPhase is of little practical value, and of theoretical interest only. As a matter of fact, its design and study seem merely to stem from analytical considerations: since one uses an independent set of measurements at each iteration, $A^{(\tau+1)}$ and z_τ are stochastically independent, a fact which considerably simplifies the convergence analysis. In stark contrast, the WF iterate uses all the samples at each iteration and thus introduces some dependencies, which makes for some delicate analysis. Overcoming these difficulties is crucial because the community is preoccupied with convergence properties of algorithms one actually runs, like error reduction (10.2.1)–(10.2.2), or would actually want to run. As we show later in Chapter 13 it is possible to develop a rigorous theory of convergence for algorithms in the style of Gerchberg-Saxton and Fienup, please see Chapter 13 for further details.

In a recent paper [164], which appeared on the arXiv preprint server as the final version of our paper [57] was under preparation, the authors explore necessary and sufficient conditions for the global convergence of an alternative minimization scheme

with generic sampling vectors. The issue is that we do not know when these conditions hold. Further, even when the algorithm converges, it does not come with an explicit convergence rate so that is is not known whether the algorithm converges in polynomial time. As before, some of our methods as well as those from our companion paper [58] may have some bearing upon the analysis of this algorithm. Similarly, another class of nonconvex algorithms that have recently been proposed in the literature are iterative algorithms based on Generalized Approximate Message Passing (GAMP), see [202] and [212] as well as [32, 34, 91] for some background literature on AMP. In [212], the authors demonstrate a favorable runtime for an algorithm of this nature. However, this does not come with any theoretical guarantees.

Moving away from the phase retrieval problem, we would like to mention some very interesting work on the matrix completion problem using non-convex schemes by Montanari and coauthors [140–142], see also [5, 25, 122, 135, 146, 180, 181]. Although the problems and models are quite different, there are some general similarities between the algorithm named OptSpace in [141] and ours. Indeed, OptSpace operates by computing an initial guess of the solution to a low-rank matrix completion problem by means of a spectral method. It then sets up a nonconvex problem, and proposes an iterative algorithm for solving it. Under suitable assumptions, [141] demonstrates the correctness of this method in the sense that OptSpace will eventually converge to a low-rank solution, although it is not shown to converge in polynomial time.

Finally, our work is completely different from other recent works on nonconvex optimization for sparse linear regression. In [155], it is shown that all local optima of a particular non-convex optimization problem are relatively close, and establish convergence of gradient schemes to a local optimum. It is an interesting research direction to see if in these regression problems, our analysis can be used to establish convergence to the global optimum.

Chapter 13

The error reduction algorithm through the lens of non-convex optimization

In this chapter we shall explain how the classic Error Reduction (ER) algorithm of Gerchberg–Saxton and Fienup can be viewed as an iterative algorithm that solves a certain non-convex optimization problem. Furthermore, we develop theory for the convergence of this algorithm starting from a sufficiently accurate initialization (which we also provide). Admittedly, these results are not as strong as their counterparts for WF as convergence is guaranteed in a smaller neighborhood of the global optimum compared with the WF results (This is in part due to time constraints of the author in the writing of this dissertation). Nevertheless, the result and theoretical analysis presented here is a first step towards providing convergence analysis of these algorithms in a larger neighborhood. Indeed, in future work the author will present a stronger convergence analysis of these algorithms that holds in a larger neighborhood. Throughout this chapter we shall assume that the signal \mathbf{x} and the measurement matrix \mathbf{A} are both real valued.

13.1 ER algorithm as non-convex optimization

To present our theoretical results let us first explain how we can view the ER algorithm through the lens of optimization.¹ For this purpose given a vector $\mathbf{v} \in \mathbb{R}^m$ define

$$f(\mathbf{v}) = \frac{1}{2} \sum_{r=1}^m (|v_r| - b_r)^2,$$

where $\mathbf{b} = |\mathbf{Ax}| \in \mathbb{R}^m$ is a vector containing the square root of our measurements ($b_r = \sqrt{y_r}$). Now consider the following optimization problem

$$\underset{\mathbf{v} \in \text{Rang}(\mathbf{A})}{\text{minimize}} \quad f(\mathbf{v}) := \frac{1}{2} \sum_{r=1}^m (|v_r| - b_r)^2 := \frac{1}{2} \|\mathbf{v} - \mathbf{b}\|_{\ell_2}^2. \quad (13.1.1)$$

We note that this is a natural optimization problem as the objective measures the misfit between our measurements and the absolute value of a vector \mathbf{v} that belongs in the range of \mathbf{A} . If we are able to find a vector \mathbf{v} that makes the objective zero (which implies that $|\mathbf{v}| = \mathbf{b}$) we can solve for the signal by solving the overdetermined system of equations $\mathbf{v} = \mathbf{Az}$.

This is an example of a constrained and non-differentiable optimization problem. There are multiple ways of solving constrained optimization problems of the form

$$\min_{\mathbf{z} \in \mathcal{Q}} f(\mathbf{z})$$

where \mathcal{Q} is the constraint set. One popular scheme is based on sub-gradient mappings defined below.

Definition 13.1.1 [186] Let us fix some $\gamma > 0$. Denote

$$\begin{aligned} \mathbf{x}_{\mathcal{Q}}(\bar{\mathbf{x}}; \gamma) &= \arg \min_{\mathbf{z} \in \mathcal{Q}} [f(\bar{\mathbf{x}}) + \langle \partial f(\bar{\mathbf{x}}), \mathbf{x} - \bar{\mathbf{x}} \rangle + \frac{\gamma}{2} \|\mathbf{x} - \bar{\mathbf{x}}\|_{\ell_2}^2] \\ g_{\mathcal{Q}}(\bar{\mathbf{x}}; \gamma) &= \gamma (\bar{\mathbf{x}} - \mathbf{x}_{\mathcal{Q}}(\bar{\mathbf{x}}; \gamma)). \end{aligned}$$

We call $g_{\mathcal{Q}}(\bar{\mathbf{x}}; \gamma)$ the sub-gradient mapping of f on \mathcal{Q} .

¹Please also see [241].

For all practical purposes we can view $g_{\mathcal{Q}}(\mathbf{v}; \gamma)$ as a sort of sub-gradient function. Therefore we can run sub-gradient descent

$$\mathbf{v}_{\tau+1} = \mathbf{v}_\tau - \mu_\tau g_{\mathcal{Q}}(\mathbf{v}_\tau; \gamma). \quad (13.1.2)$$

We shall use $\gamma = 1$ (this choice becomes more clear in the proofs). After simple calculations, (13.1.2) reduces to

$$\mathbf{v}_{\tau+1} = (1 - \mu_\tau)\mathbf{v}_\tau - \mu_\tau \mathcal{P}_A(\mathbf{v}_\tau - \partial f(\mathbf{v}_\tau)). \quad (13.1.3)$$

Here, $\mathcal{P}_A(\mathbf{v}) = \mathbf{A}^* (\mathbf{A}^* \mathbf{A})^{-1} \mathbf{A} \mathbf{v}$ is the projection of \mathbf{v} onto the range of the matrix \mathbf{A} . By a simple calculation $\partial f(\mathbf{v}) = (|\mathbf{v}| - \mathbf{b}) \odot \frac{\mathbf{v}}{|\mathbf{v}|}$. Now let us use the iteration in (13.1.3) with $\mu_\tau = 1$ we have

$$\mathbf{v}_{\tau+1} = \mathcal{P}_A \left(\mathbf{b} \odot \frac{\mathbf{v}_\tau}{|\mathbf{v}_\tau|} \right).$$

Comparing the latter with (10.2.3) we can see that this is exactly the ER update!

13.2 Some theory for the convergence of the ER algorithm

In this section we shall present our theoretical results. In this result we assume that the sampling vectors follow the real Gaussian model where $\mathbf{a}_r \in \mathbb{R}^n \stackrel{\text{i.i.d.}}{\sim} \mathcal{N}(\mathbf{0}, \mathbf{I})$.

Theorem 13.2.1 *Let \mathbf{x} be a fixed vector in \mathbb{R}^n and $\mathbf{A} \in \mathbb{R}^{m \times n}$ be a Gaussian sensing matrix with entries i.i.d. $\mathcal{N}(0, 1)$. Also, let $\mathbf{y} = |\mathbf{A}\mathbf{x}|^2 \in \mathbb{R}^m$ be m quadratic samples with $m \geq c_0 \cdot n(\log n)^2$, where c_0 is a sufficiently large numerical constant. Let $\mathbf{v}_0 = \mathbf{A}\mathbf{z}_0$ where \mathbf{z}_0 is the output of Algorithm 8. Then, \mathbf{v}_0 obeys*

$$\text{dist}(\mathbf{v}_0, \mathbf{A}\mathbf{x}) < \frac{\sqrt{m} - \sqrt{n}}{\sqrt{8n}} \min_r |\mathbf{a}_r^* \mathbf{x}| \quad (13.2.1)$$

with probability at least $1 - 12e^{-\gamma n} - 8/n^2$ (γ is a fixed positive numerical constant).

Further, take a constant learning parameter sequence, $\mu_\tau = \mu$ for all $\tau = 1, 2, \dots$ and assume $\mu \leq 1$. Then there is an even of probability at least $1 - m \left(\frac{e}{2}\right)^{-\frac{n}{2}} - 2^{-n} - 2e^{-n}$, such that on this event, starting from any initial solution \mathbf{v}_0 obeying (13.2.1), we have

$$\text{dist}(\mathbf{v}_\tau, \mathbf{Ax}) \leq \frac{\sqrt{m} - \sqrt{n}}{\sqrt{4\pi n}} (1 - \mu)^{\tau/2} \cdot \|\mathbf{x}\|_{\ell_2}. \quad (13.2.2)$$

Here, \mathbf{v}_τ is obtained via the update defined by (13.1.3).

As stated previously the ER update is a special case of the updates of the theorem above with $\mu = 1$. Therefore, the above theorem shows that the ER update (starting from a sufficiently good initialization) converges with a geometric rate to \mathbf{Ax} . We note that if convergence to the signal is of interest we can set $\mathbf{z}_\tau = (\mathbf{A}^* \mathbf{A})^{-1} \mathbf{A}^* \mathbf{v}_\tau$ and show that with high probability

$$\text{dist}(\mathbf{z}_\tau, \mathbf{x}) \leq \frac{1}{\sqrt{\pi n}} (1 - \mu)^{\tau/2} \cdot \|\mathbf{x}\|_{\ell_2}.$$

The latter expression was obtained from (13.2.2) by using well-known lower bounds on the singular value of a random Gaussian matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ with entries i.i.d. $\mathcal{N}(0, 1)$.

Chapter 14

Numerical experiments

In this chapter we present some numerical experiments to assess the empirical performance of PhaseLift and Wirtinger flow algorithms and their stability to noise.

14.1 Models

Before presenting the results we introduce the signal and measurement models we use throughout this chapter.

14.1.1 Signal models

We consider two signal models:

- *Random low-pass signals.* Here, \mathbf{x} is given by

$$x[t] = \sum_{k=-(M/2-1)}^{M/2} (X_k + iY_k) e^{2\pi i(k-1)(t-1)/n},$$

with $M = n/8$ and X_k and Y_k are i.i.d. $\mathcal{N}(0, 1)$.

- *Random Gaussian signals.* In this model, $\mathbf{x} \in \mathbb{C}^n$ is a random complex Gaussian vector with i.i.d. entries of the form $x[t] = X + iY$ with X and Y distributed as

$\mathcal{N}(0, 1)$; this can be expressed as

$$x[t] = \sum_{k=-(n/2-1)}^{n/2} (X_k + iY_k) e^{2\pi i(k-1)(t-1)/n},$$

where X_k and Y_k are i.i.d. $\mathcal{N}(0, 1/8)$ so that the low-pass model is a ‘bandlimited’ version of this high-pass random model (variances are adjusted so that the expected power is the same).

14.1.2 Measurement models

We perform simulations based on four different kinds of measurements:

- *Gaussian measurements.* We sample $m = nL$ random complex Gaussian vectors \mathbf{a}_k and use measurements of the form $|\mathbf{a}_k^* \mathbf{x}|^2$.
- *Binary modulations/codes.* We sample $(L - 1)$ binary codes distributed as

$$d = \begin{cases} 1 & \text{with prob. } \frac{1}{2} \\ 0 & \text{with prob. } \frac{1}{2} \end{cases}$$

together with a regular diffraction pattern ($d[t] = 1$ for all t).

- *Ternary modulations/codes.* We sample $(L - 1)$ ternary codes distributed as (11.3.3) together with a regular diffraction pattern.
- *Octanary modulations/codes.* Here, the codes are distributed as (11.3.2).

14.2 Synthetic experiments

In this section we assess the performance of both PhaseLift and Wirtinger Flow as well as their stability on synthetic examples. Due to computational issues for PhaseLift we shall only focus on the CDP model, while for Wirtinger Flow we shall show results for both the Gaussian and CDP models.

14.2.1 Phase transition of Phase Lift using CDP measurements

We carry out some numerical experiments to show how the performance of the algorithm depends on the number of measurements/coded patterns. For this purpose we consider 50 trials. In each trial we generate a random complex vector $\mathbf{x} \in \mathbb{C}^n$ (with $n = 128$) from both signal models and gather data according to the four different measurement models above. For each trial we solve the following optimization problem

$$\min \quad \frac{1}{2} \|\mathbf{b} - \mathcal{A}(\mathbf{X})\|_{\ell_2}^2 + \lambda \text{tr}(\mathbf{X}) \quad \text{subject to} \quad \mathbf{X} \succeq \mathbf{0} \quad (14.2.1)$$

with $\lambda = 10^{-3}$ (Note that the solution to (14.2.1) as λ tends to zero will equal to the optimal solution of (11.1.2)).

In Figure 14.1 we report the empirical probability of success for different signal and measurement models with different number of measurements. We declare a trial successful if the relative error of the reconstruction ($\|\hat{\mathbf{X}} - \mathbf{x}\mathbf{x}^*\|_F / \|\mathbf{x}\mathbf{x}^*\|_F$) falls below 10^{-5}). These plots suggest that for the type of models studied in this paper six coded patterns are sufficient for exact recovery via convex programming.

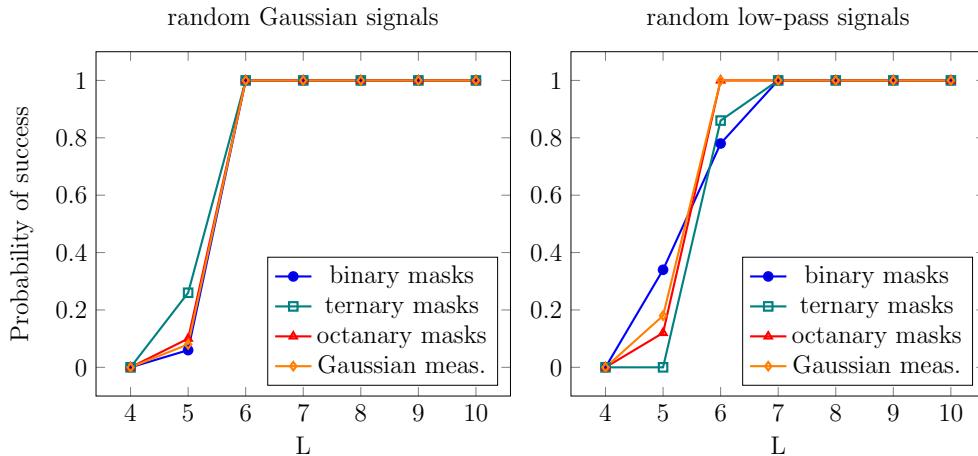


Figure 14.1: Empirical probability of success based on 50 random trials for different signal/measurement models and a varied number of measurements. A value of L on the x-axis means that we have a total of $m = Ln$ samples.

14.2.2 Phase transition of Wirtinger Flow using Gaussian and CDP measurements

In this section we examine the performance of the Wirtinger flow algorithm for recovering random signals $\mathbf{x} \in \mathbb{C}^n$ under the Gaussian and coded diffraction models. Below, we set $n = 128$, and generate one signal of each type which will be used in all the experiments.

The initialization step of the Wirtinger flow algorithm is run by applying 50 iterations of the power method outlined in Algorithm 9 from Appendix I. In the iteration (12.1.2), we use the parameter value $\mu_\tau = \min(1 - \exp(-\tau/\tau_0), 0.2)$ where $\tau_0 \approx 330$. We stop after 2,500 iterations, and report the empirical probability of success for the two different signal models. The empirical probability of success is an average over 100 trials, where in each instance, we generate new random sampling vectors according to the Gaussian or CDP models. We declare a trial successful if the relative error of the reconstruction $\text{dist}(\hat{\mathbf{x}}, \mathbf{x}) / \|\mathbf{x}\|_{\ell_2}$ falls below 10^{-5} .

Figure 14.2 shows that around $4.5n$ Gaussian phaseless measurements suffice for exact recovery with high probability via the Wirtinger flow algorithm. We also see that about six octanary patterns are sufficient.

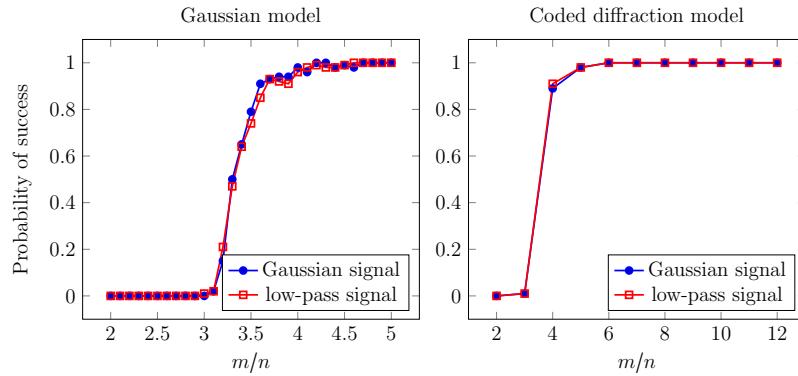


Figure 14.2: Empirical probability of success based on 100 random trials for different signal/measurement models and a varied number of measurements. The coded diffraction model uses octanary patterns; the number of patterns $L = m/n$ only takes on integral values.

14.2.3 Noisy measurements

We now study how the performance of different variations of the convex relaxation (PhaseLift) and the WF algorithm in the presence of noise. We consider Poisson noise which is the usual noise model in optics. More, specifically we assume that the measurements $\{y_r\}_{r=1}^m$ is a sequence of independent samples from the Poisson distributions $\text{Poi}(\eta_r)$, where $\eta_r = |\mathbf{a}_r^* \mathbf{x}|^2$ correspond to the noiseless measurements. The Poisson log-likelihood for independent samples has the form $\sum_r y_r \log \eta_r - \eta_r$ (up to an additive constant factor).

In our experiments, the test signal is again a complex random signal sampled according to the two models described in Section 14.1.1. We use eight CDP's according to the three models described in Section 14.1.2. Poisson noise is adjusted so that the SNR levels range from 10 to 50dB. Here, $\text{SNR} = \|\mathcal{A}(\mathbf{x}\mathbf{x}^*)\|_{\ell_2} / \|\mathbf{b} - \mathcal{A}(\mathbf{x}\mathbf{x}^*)\|_{\ell_2}$ is the signal-to-noise ratio. For each SNR level we repeat the experiment ten times with different random noise and different random CDP's.

We consider four different approaches:

- *Convex relaxation with regularization (Reg CVX)*. Following a classical fitting approach we balance a maximum likelihood term with the trace norm in the relaxation (11.1.2):

$$\min \sum_r [\eta_r - y_r \log \eta_r] + \lambda \text{tr}(\mathbf{X}) \quad \text{subject to} \quad \boldsymbol{\eta} = \mathcal{A}(\mathbf{X}) \quad \text{and} \quad \mathbf{X} \succeq \mathbf{0}.$$

For the regularization parameter we use $\lambda = 1/\text{SNR}$. (In these experiments, the value of SNR is known. The result, however, is rather insensitive to the choice of the parameter λ and a good choice for the regularization parameter λ can be obtained by cross validation.)

- *Convex relaxation without Regularization (CVX no Reg)*. This approach is similar to the previous algorithm. However, we do not add in the regularization of

the trace in the relaxation (11.1.2):

$$\min \sum_r [\eta_r - y_r \log \eta_r] \quad \text{subject to} \quad \boldsymbol{\eta} = \mathcal{A}(\mathbf{X}) \quad \text{and} \quad \mathbf{X} \succeq \mathbf{0}.$$

- *Convex relaxation with ℓ_1 fit (ℓ_1 fit).* Again we use the relaxation in (11.1.2). However, this time instead of the Poisson likelihood we use an ℓ_1 fit to the noisy measurements:

$$\min \sum_r |\eta_r - y_r| \quad \text{subject to} \quad \boldsymbol{\eta} = \mathcal{A}(\mathbf{X}) \quad \text{and} \quad \mathbf{X} \succeq \mathbf{0}.$$

- *Wirtinger Flow with Poisson likelihood (WF-Poisson).* In this approach we use the exact same initialization as WF. However, for our iterate updates we use the gradient (based on Wirtinger derivatives) of the following non-convex objective in lieu of the gradient update of WF

$$f(\mathbf{z}) = \frac{1}{2m} \sum_{r=1}^m (|\mathbf{a}_r^* \mathbf{z}|^2 - 2y_r \log |\mathbf{a}_r^* \mathbf{z}|).$$

For the convex schemes since the optimal solution may not be rank one we find the best rank one approximation $\hat{\mathbf{x}}\hat{\mathbf{x}}^*$ to the optimal solution of the convex problem. Figure 14.3 shows the average relative MSE (in dB) versus the SNR (also in dB) for different algorithms. More precisely, the values of $10 \log_{10}(\text{rel. MSE})$ are plotted, where $\text{rel. MSE} = \|\hat{\mathbf{x}}\hat{\mathbf{x}}^* - \mathbf{x}\mathbf{x}^*\|_F^2 / \|\mathbf{x}\mathbf{x}^*\|_F^2$. These figures indicate that the performance of the algorithm degrades linearly as the SNR decreases (on a dB/dB scale). Empirically, the slope is close to -1, which means that the MSE scales like the noise. Together with the low offset, these features indicate that all is as in a well-conditioned-least squares problem. Interestingly WF despite having significantly lower computational cost, significantly out performs its convex counterparts.

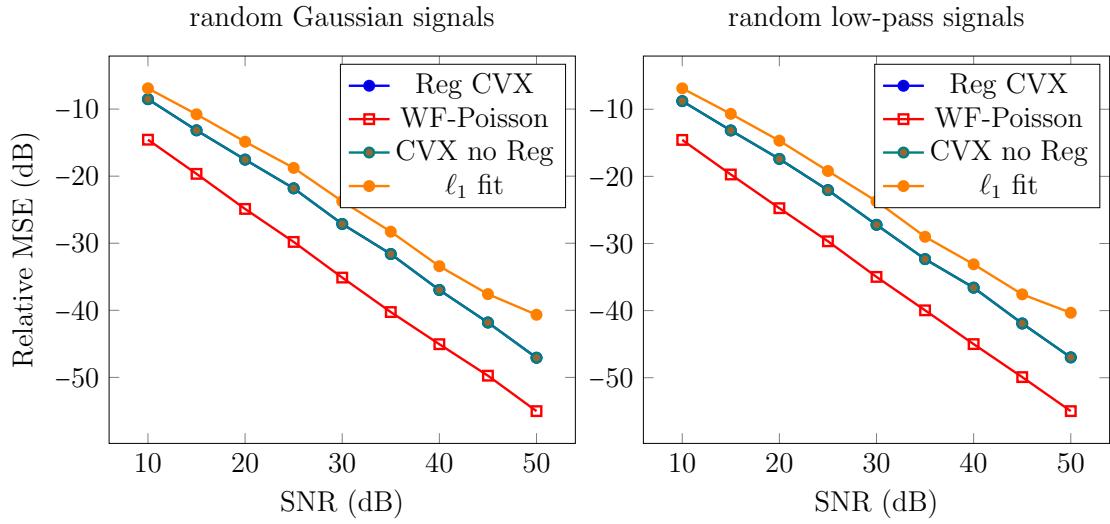


Figure 14.3: SNR versus relative MSE on a dB-scale for different kinds of signal/measurement models and algorithms. The linear relationship between SNR and MSE (on the dB scale) is apparent. The MSE behaves as in a well-conditioned least-squares problem.

14.3 Performance on natural images

We move on to testing the Wirtinger flow algorithm on various images of different sizes; these are photographs of the Naqsh-e Jahan Square in the central Iranian city of Esfahan, the Stanford main quad, and the Milky Way galaxy. Since each photograph is in color, we run the WF algorithm on each of the three RGB images that make up the photograph. Color images are viewed as $n_1 \times n_2 \times 3$ arrays, where the first two indices encode the pixel location, and the last the color band.

We generate $L = 20$ random octanary patterns and gather the coded diffraction patterns for each color band using these 20 samples. As before, we run 50 iterations of the power method as the initialization step. The updates use the sequence $\mu_\tau = \min(1 - \exp(-\tau/\tau_0), 0.4)$ where $\tau_0 \approx 330$ as before. In all cases we run 300 iterations and record the relative recovery error as well as the running time. If \mathbf{x} and $\hat{\mathbf{x}}$ are the original and recovered images, the relative error is equal to $\|\hat{\mathbf{x}} - \mathbf{x}\|_{\ell_2} / \|\mathbf{x}\|_{\ell_2}$, where $\|\cdot\|_{\ell_2}$ is the Euclidean norm $\|\mathbf{x}\|_{\ell_2}^2 = \sum_{i,j,k} |x(i, j, k)|^2$. The computational time

we report is the computational time averaged over the three RGB images. All experiments were carried out on a MacBook Pro with a 2.4 GHz Intel Core i7 Processor and 8 GB 1600 MHz DDR3 memory.

Figure 14.4 shows the images recovered via the Wirtinger flow algorithm. In all cases, WF gets 12 or 13 digits of precision in a matter of minutes. To convey an idea of timing that is platform-independent, we also report time in units of FFTs; one FFT unit is the amount of time it takes to perform a single FFT on an image of the same size. Now all the workload is dominated by matrix vector products of the form $\mathbf{A}\mathbf{z}$ and $\mathbf{A}^*\mathbf{v}$. In details, each iteration of the power method in the initialization step, or each update (12.1.2) requires 40 FFTs; the factor of 40 comes from the fact that we have 20 patterns and that each iteration involves one FFT and one adjoint or inverse FFT. Hence, the total number of FFTs is equal to

$$20 \text{ patterns} \times 2 \text{ (one FFT and one IFFT)} \times (300 \text{ gradient steps} + 50 \text{ power iterations}) = 14,000.$$

Another way to state this is that the total workload of our algorithm is roughly equal to 350 applications of the sensing matrix \mathbf{A} and its adjoint \mathbf{A}^* . For about 13 digits of accuracy (relative error of about 10^{-13}), Figure 14.4 shows that we need between 21,000 and 42,000 FFT units. This is within a factor between 1.5 and 3 of the optimal number computed above. This increase has to do with the fact that in our implementation, certain variables are copied into other temporary variables and these types of operations cause some overhead. This overhead is non-linear and becomes more prominent as the size of the signal increases.

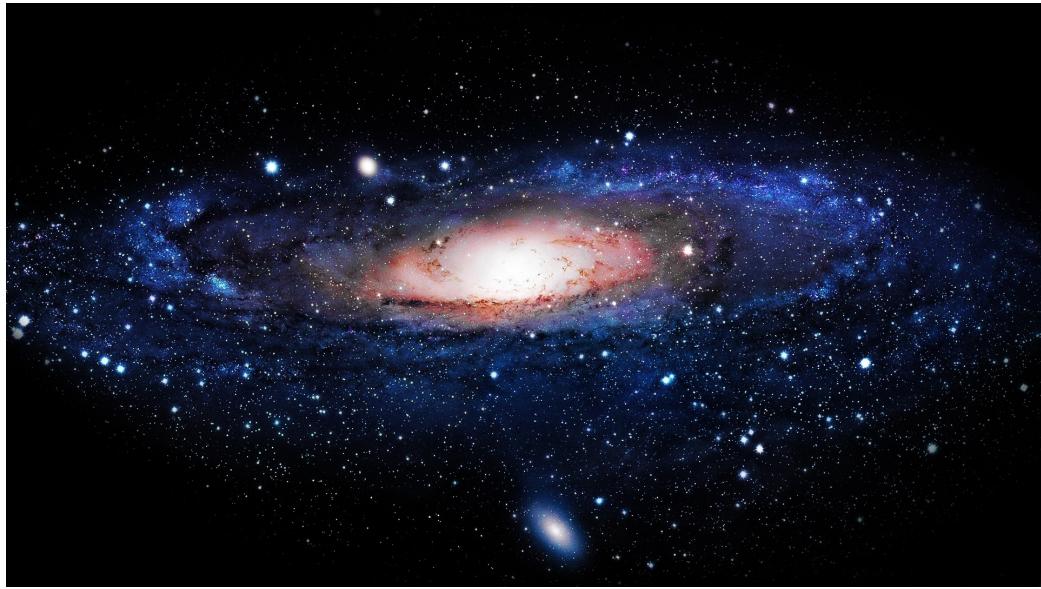
For comparison, SDP based solutions such as PhaseLift [52,64] and PhaseCut [237] would be prohibitive on a laptop computer as the lifted signal would not fit into memory. In the SDP approach an n pixel image become an $n^2/2$ array, which in the first example already takes storing the lifted signal even for the smallest image requires $(189 \times 768)^2 \times 1/2 \times 8$ Bytes, which is approximately 85 GB of space. (For the image of the Milky Way, storage would be about 17 TB.) These large memory requirements prevent the application of full-blown SDP solvers on desktop computers.



(a) Naqsh-e Jahan Square, Esfahan. Image size is 189×768 pixels; timing is 61.4 sec or about 21,200 FFT units. The relative error is 6.2×10^{-16} .



(b) Stanford main quad. Image size is 320×1280 pixels; timing is 181.8120 sec or about 20,700 FFT units. The relative error is 3.5×10^{-14} .



(c) Milky way Galaxy. Image size is 1080×1920 pixels; timing is 1318.1 sec or 41,900 FFT units. The relative error is 9.3×10^{-16} .

Figure 14.4: Performance of the WF algorithm on three scenic images. Image size, computational time in seconds and in units of FFTs are reported, as well as the relative error after 300 WF iterations.

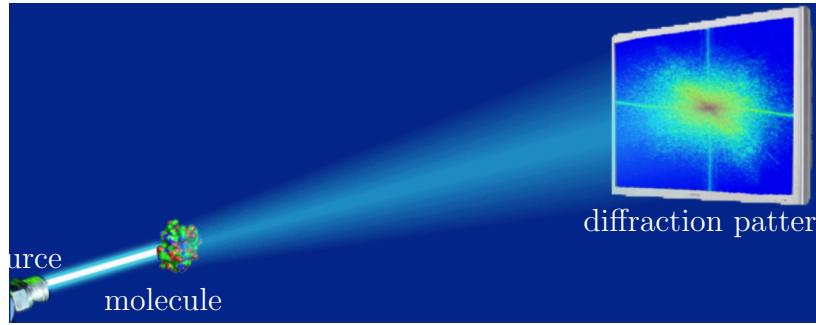


Figure 14.5: An illustrative setup of diffraction patterns.

14.4 3D molecules

Understanding molecular structure is a great contemporary scientific challenge, and several techniques are currently employed to produce 3D images of molecules; these include electron microscopy and X-ray imaging. In X-ray imaging, for instance, the experimentalist illuminates an object of interest, e.g. a molecule, and then collects the intensity of the diffracted rays, please see Figure 14.5 for an illustrative setup. Figures 14.6 and 14.7 show the schematic representation and the corresponding electron density maps for the Caffeine and Nicotine molecules: the density map $\rho(x_1, x_2, x_3)$ is the 3D object we seek to infer. In this paper, we do not go as far 3D reconstruction but demonstrate the performance of the Wirtinger flow algorithm for recovering projections of 3D molecule density maps from simulated data. For related simulations using convex schemes we refer the reader to [109].

To derive signal equations, consider an experimental apparatus as in Figure 14.5. If we imagine that light propagates in the direction of the x_3 -axis, an approximate model for the collected data reads

$$I(f_1, f_2) = \left| \int \left(\int \rho(x_1, x_2, x_3) \chi_3 \right) e^{-2i\pi(f_1 x_1 + f_2 x_2)} \chi_1 \chi_2 \right|^2.$$

In other words, we collect the intensity of the diffraction pattern of the projection $\int \rho(x_1, x_2, x_3) \chi_3$. The 2D image we wish to recover is thus the line integral of the density map along a given direction. As an example, the Caffeine molecule along with

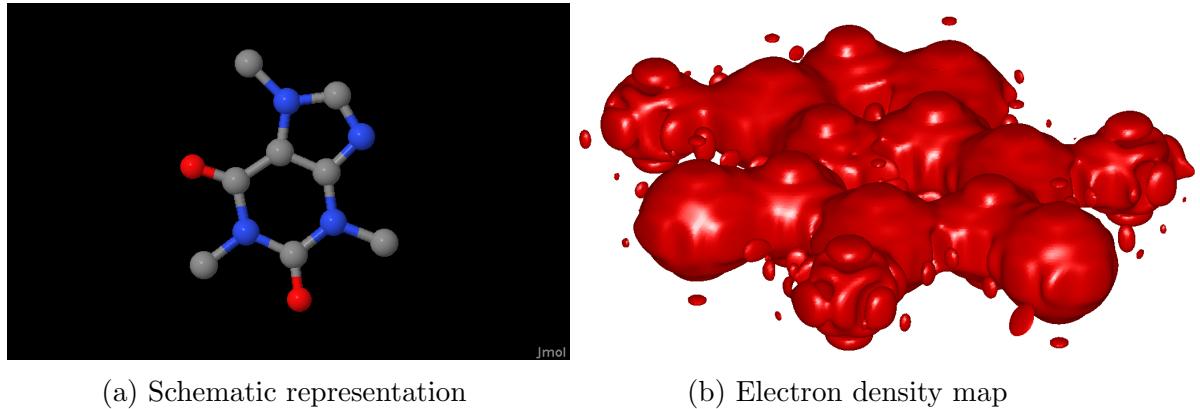


Figure 14.6: Schematic representation and electron density map of the Caffeine molecule.

its projection on the x_1x_2 -plane (line integral in the x_3 direction) is shown in Figure 14.8. Now, if we let R be the Fourier transform of the density ρ , one can re-express the identity above as

$$I(f_1, f_2) = |R(f_1, f_2, 0)|^2.$$

Therefore, by imputing the missing phase using phase retrieval algorithms, one can recover a slice of the 3D Fourier transform of the electron density map, i.e. $R(f_1, f_2, 0)$. Viewing the object from different angles or directions gives us different slices. In a second step we do not perform in this paper, one can presumably recover the 3D Fourier transform of the electron density map from all these slices (this is the tomography or blind tomography problem depending upon whether or not the projection angles are known) and, in turn, the 3D electron density map.

We now generate 51 observation planes by rotating the x_1x_2 -plane around the x_1 -axis by equally spaced angles in the interval $[0, 2\pi]$. Each of these planes is associated with a 2D projection of size 1024×1024 , giving us 20 coded diffraction octanary patterns (we use the same patterns for all 51 projections). We run the Wirtinger flow algorithm with exactly the same parameters as in the previous section, and stop after 150 gradient iterations. Figure 14.9 reports the average relative error over the 51 projections and the total computational time required for reconstructing all 51 images.

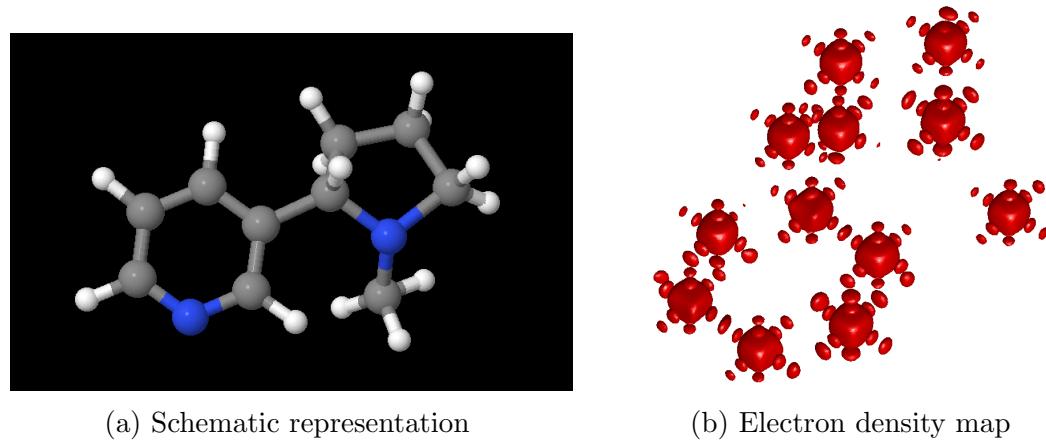


Figure 14.7: Schematic representation and electron density map of the Nicotine molecule.

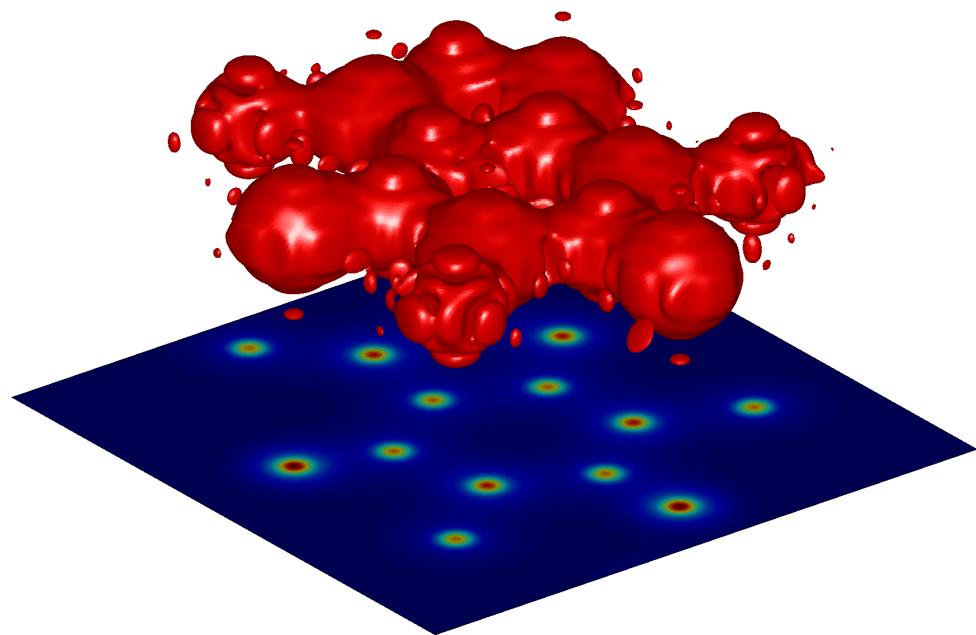


Figure 14.8: Electron density $\rho(x_1, x_2, x_3)$ of the Caffeine molecule along with its projection onto the x_1x_2 -plane.

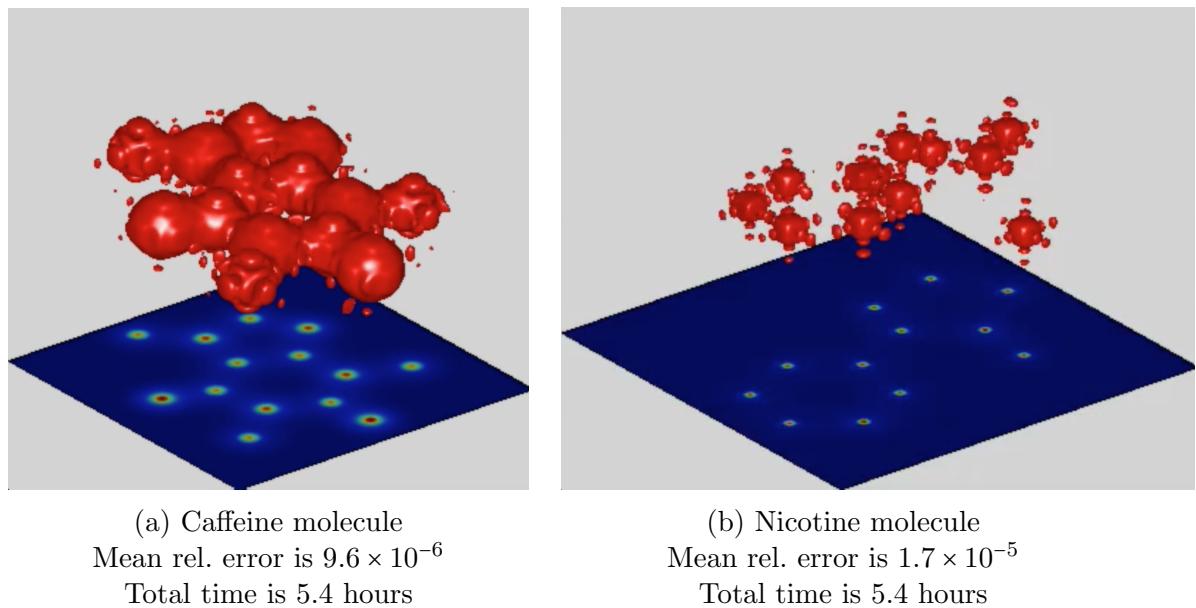


Figure 14.9: Reconstruction sequence of the projection of the Caffeine and Nicotine molecules along different directions. To see the videos please visit the author's website.

Chapter 15

Proofs

15.1 Proofs for PhaseLift with CDP measurements

In this section we prove our results concerning the exactness of PhaseLift with CDP measurements. Before we begin, we introduce some notation. We recall that the random variable d is admissible, i.e. bounded i.e. $|d| \leq M$, symmetric, and obeying moment constraints

$$\mathbb{E} d = 0, \quad \mathbb{E} d^2 = 0, \quad \mathbb{E} |d|^4 = 2(\mathbb{E} |d|^2)^2. \quad (15.1.1)$$

Without loss of generality we also assume that $\mathbb{E} |d|^2 = 1$. Throughout \mathbf{D} is a diagonal matrix with i.i.d. entries distributed as d . For a vector $\mathbf{y} \in \mathbb{C}^n$ we use \mathbf{y}^T and \mathbf{y}^* to denote the transpose and complex conjugate of the vector \mathbf{y} . We also use $\bar{\mathbf{y}}$ to denote elementwise conjugation of the entries of \mathbf{y} . Since this is less standard, we prefer to be concrete as to avoid ambiguity: for example,

$$\begin{bmatrix} 1+i \\ 1+2i \end{bmatrix}^T = \begin{bmatrix} 1+i & 1+2i \end{bmatrix}, \quad \begin{bmatrix} 1+i \\ 1+2i \end{bmatrix}^* = \begin{bmatrix} 1-i & 1-2i \end{bmatrix}, \quad \overline{\begin{bmatrix} 1+i \\ 1+2i \end{bmatrix}} = \begin{bmatrix} 1-i \\ 1-2i \end{bmatrix}.$$

Continuing, $\|\mathbf{X}\|$ is the spectral or operator norm of a matrix \mathbf{X} . Finally, $\mathbf{1}$ is a vector with all entries equal to one.

Throughout, we assume that the fixed vector \mathbf{x} we seek to recover is unit normed,

i.e. $\|\mathbf{x}\|_{\ell_2} = 1$. Throughout T is the linear subspace

$$T = \{\mathbf{X} = \mathbf{x}\mathbf{y}^* + \mathbf{y}\mathbf{x}^* : \mathbf{y} \in \mathbb{C}^n\}.$$

This subspace may be interpreted as the tangent space at $\mathbf{x}\mathbf{x}^*$ to the manifold of Hermitian matrices of rank 1. Below T^\perp is the orthogonal complement to T . For a linear subspace V of Hermitian matrices, we use \mathbf{Y}_V or $\mathcal{P}_V(\mathbf{Y})$ to denote the orthogonal projection of \mathbf{Y} onto V . With this, the reader will check that $\mathbf{Y}_{T^\perp} = (\mathbf{I} - \mathbf{x}\mathbf{x}^*)\mathbf{Y}(\mathbf{I} - \mathbf{x}\mathbf{x}^*)$.

15.1.1 Preliminaries

It is useful to record two identities that shall be used multiple times, and defer the proofs to the Appendix E.

Lemma 15.1.1 *For any fixed vector $\mathbf{x} \in \mathbb{C}^n$*

$$\mathbb{E}\left(\frac{1}{nL}\mathcal{A}^*\mathcal{A}(\mathbf{x}\mathbf{x}^*)\right) = \mathbb{E}\left(\frac{1}{n}\sum_{k=1}^n |\mathbf{f}_k^*\mathbf{D}^*\mathbf{x}|^2 \mathbf{D}\mathbf{f}_k\mathbf{f}_k^*\mathbf{D}^*\right) = \mathbf{x}\mathbf{x}^* + \|\mathbf{x}\|_{\ell_2}^2 \mathbf{I}.$$

Lemma 15.1.2 *For any fixed $\mathbf{x} \in \mathbb{C}^n$,*

$$\mathbb{E}\left(\frac{1}{n}\sum_{k=1}^n (\mathbf{f}_k^*\mathbf{D}^*\mathbf{x})^2 \mathbf{D}\mathbf{f}_k\mathbf{f}_k^T\mathbf{D}\right) = 2\mathbf{x}\mathbf{x}^T.$$

Next, we present two simple intermediate results we shall also use. The proofs are also in the Appendix E.

Lemma 15.1.3 *Fix $\delta > 0$ and suppose the number L of CDP's obeys $L \geq c \log n$ for some sufficiently large numerical constant c . Then with probability at least $1 - 1/n^2$,*

$$\left\| \frac{1}{nL}\mathcal{A}^*(\mathbf{1}) - \mathbf{I}_n \right\| \leq \delta.$$

Lemma 15.1.4 *For all positive semidefinite matrices \mathbf{X} , it holds*

$$\frac{1}{nL} \|\mathcal{A}(\mathbf{X})\|_{\ell_1} \leq M^2 \text{trace}(\mathbf{X}).$$

Finally, the last piece of mathematics is the matrix Hoeffding inequality

Lemma 15.1.5 [226, Theorem 1.3] Let $\{S_\ell\}_{\ell=1}^L$ be a sequence of independent random $n \times n$ self-adjoint matrices. Assume that each random matrix obeys

$$\mathbb{E} S_\ell = \mathbf{0} \quad \text{and} \quad \|S_\ell\| \leq \Delta \quad \text{almost surely.} \quad (15.1.2)$$

Then for all $t \geq 0$,

$$\mathbb{P}\left(\frac{1}{L} \left\| \sum_{\ell=1}^L S_\ell \right\| \geq t\right) \leq 2n \exp\left(-\frac{Lt^2}{8\Delta^2}\right). \quad (15.1.3)$$

15.1.2 Certificates

We now establish sufficient conditions guaranteeing that $\mathbf{x}^* \mathbf{x}$ is the unique feasible point of (11.2.3). Variants of the lemma below have appeared before in the literature, see [55, 64, 82].

Lemma 15.1.6 Suppose the mapping \mathcal{A} obeys the following two properties:

1. For all matrices $\mathbf{X} \in T$

$$\frac{1}{\sqrt{nL}} \|\mathcal{A}(\mathbf{X})\|_{\ell_2} \geq \frac{(1-\delta)}{\sqrt{2}} \|\mathbf{X}\|_F. \quad (15.1.4)$$

2. There exists a self-adjoint matrix of the form $\mathbf{Z} = \mathcal{A}^*(\boldsymbol{\lambda})$, with $\boldsymbol{\lambda}$ real valued (this makes sure that \mathbf{Z} is self adjoint), obeying

$$\mathbf{Z}_{T^\perp} \leq -\mathbf{I}_{T^\perp} \quad \text{and} \quad \|\mathbf{Z}_T\|_F \leq \frac{1-\delta}{2M^2\sqrt{nL}}. \quad (15.1.5)$$

Then $\mathbf{x}^* \mathbf{x}$ is the unique element in the feasible set (11.2.3).

Proof Suppose $\mathbf{x} \mathbf{x}^* + \mathbf{H}$ is feasible. Feasibility implies that \mathbf{H} is a self-adjoint matrix in the null space of \mathcal{A} and $\mathbf{H}_{T^\perp} \geq 0$. This is because for all $\mathbf{y} \perp \mathbf{x}$,

$$\mathbf{y}^* (\mathbf{x} \mathbf{x}^* + \mathbf{H}) \mathbf{y} = \mathbf{y}^* \mathbf{H} \mathbf{y} \geq 0,$$

which says that \mathbf{H}_{T^\perp} is positive semidefinite. This gives

$$\langle \mathbf{H}, \mathbf{Y} \rangle = 0 = \langle \mathbf{H}_T, \mathbf{Z}_T \rangle + \langle \mathbf{H}_{T^\perp}, \mathbf{Z}_{T^\perp} \rangle.$$

On the one hand,

$$\langle \mathbf{H}_T, \mathbf{Z}_T \rangle = -\langle \mathbf{H}_{T^\perp}, \mathbf{Z}_{T^\perp} \rangle \geq \langle \mathbf{H}_{T^\perp}, \mathbf{I}_{T^\perp} \rangle = \text{trace}(\mathbf{H}_{T^\perp}). \quad (15.1.6)$$

Therefore,

$$\text{trace}(\mathbf{H}_{T^\perp}) \geq \frac{1}{M^2 n L} \|\mathcal{A}(\mathbf{H}_{T^\perp})\|_{\ell_1} \geq \frac{1}{M^2 n L} \|\mathcal{A}(\mathbf{H}_{T^\perp})\|_{\ell_2}.$$

where the first inequality above follows from Lemma 15.1.4. The injectivity property (15.1.4) gives

$$\frac{1}{\sqrt{nL}} \|\mathcal{A}(\mathbf{H}_T)\|_{\ell_2} \geq \frac{(1-\delta)}{\sqrt{2}} \|\mathbf{H}_T\|_F$$

and since $\mathcal{A}(\mathbf{H}_T) = -\mathcal{A}(\mathbf{H}_{T^\perp})$, we established

$$\text{trace}(\mathbf{H}_{T^\perp}) \geq \frac{1-\delta}{\sqrt{2nL} M^2} \|\mathbf{H}_T\|_F. \quad (15.1.7)$$

On the other hand,

$$|\langle \mathbf{H}_T, \mathbf{Z}_T \rangle| \leq \|\mathbf{H}_T\|_F \|\mathbf{Z}_T\|_F \leq \frac{1-\delta}{2M^2 \sqrt{nL}} \|\mathbf{H}_T\|_F. \quad (15.1.8)$$

In summary, (15.1.6), (15.1.7) and (15.1.8) assert that $\mathbf{H}_T = 0$. In turn, this gives $\text{trace}(\mathbf{H}_{T^\perp}) = 0$ by (15.1.6), which implies that $\mathbf{H}_{T^\perp} = \mathbf{0}$ since $\mathbf{H}_{T^\perp} \succeq \mathbf{0}$. This completes the proof. ■

Property (15.1.4) can be viewed as a form of robust injectivity of the mapping \mathcal{A} restricted to elements in T . It is of course reminiscent of the local restricted isometry property in compressive sensing. Property (15.1.5) can be interpreted as the existence of an approximate dual certificate. It is well known that injectivity together with an exact dual certificate leads to exact reconstruction. The above lemma essentially

asserts that a robust form of injectivity together with an approximate dual certificate leads to exact recovery as in [64, Section 2.1], see also [119]. In the next two sections we show that the two properties stated in Lemma 15.1.6 above each hold with probability at least $1 - 1/(2n)$.

15.1.3 Robust injectivity

Lemma 15.1.7 *Fix $\delta > 0$ and suppose L obeys $L \geq c \log^3 n$ for some sufficiently large numerical constant c . Then with probability at least $1 - 1/2n$, for all $\mathbf{X} \in T$,*

$$\frac{1}{\sqrt{nL}} \|\mathcal{A}(\mathbf{X})\|_{\ell_2} \geq \frac{(1-\delta)}{\sqrt{2}} \|\mathbf{X}\|_F.$$

Proof First, notice that without loss of generality we can assume that $\mathbf{x}^* \mathbf{y}$ is real valued in the definition of T . That is,

$$T = \{\mathbf{X} = \mathbf{x}\mathbf{y}^* + \mathbf{y}\mathbf{x}^* : \mathbf{y} \in \mathbb{C}^n \text{ and } \mathbf{x}^* \mathbf{y} \in \mathbb{R}\}.$$

The reason why this is true is that for any $\mathbf{y} \in \mathbb{C}^n$, we can find $\lambda \in \mathbb{R}$, such that $\mathbf{x}^* \mathbf{y} - i\lambda \mathbf{x}^* \mathbf{x} = \mathbf{x}^* (\mathbf{y} - i\lambda \mathbf{x}) \in \mathbb{R}$ while

$$\mathbf{x}(\mathbf{y} - i\lambda \mathbf{x})^* + (\mathbf{y} - i\lambda \mathbf{x})\mathbf{x}^* = \mathbf{x}\mathbf{y}^* + \mathbf{y}\mathbf{x}^*,$$

Now for any $\mathbf{X} = \mathbf{x}\mathbf{y}^* + \mathbf{y}\mathbf{x}^* \in T$,

$$\|\mathbf{X}\|_F = \|\mathbf{x}\mathbf{y}^* + \mathbf{y}\mathbf{x}^*\|_F \leq \|\mathbf{x}\mathbf{y}^*\|_F + \|\mathbf{y}\mathbf{x}^*\|_F \leq 2 \|\mathbf{x}\|_{\ell_2} \|\mathbf{y}\|_{\ell_2} = 2 \|\mathbf{y}\|_{\ell_2}, \quad (15.1.9)$$

where we recall that $\|\mathbf{x}\|_{\ell_2} = 1$. Hence, it suffices to show that

$$\frac{1}{\sqrt{nL}} \|\mathcal{A}(\mathbf{x}\mathbf{y}^* + \mathbf{y}\mathbf{x}^*)\|_{\ell_2} \geq \frac{(1-\delta)}{\sqrt{2}} \|\mathbf{y}\|_{\ell_2}. \quad (15.1.10)$$

We have

$$\|\mathcal{A}(\mathbf{x}\mathbf{y}^* + \mathbf{y}\mathbf{x}^*)\|_{\ell_2}^2 = \sum_{\ell=1}^L \sum_{k=1}^n \left(\mathbf{f}_k^* \mathbf{D}_\ell^* (\mathbf{x}\mathbf{y}^* + \mathbf{y}\mathbf{x}^*) \mathbf{D}_\ell \mathbf{f}_k \right)^2.$$

(The reader might have expected a sum of squared moduli but since $\mathbf{xy}^* + \mathbf{yx}^*$ is self adjoint, $\mathbf{f}_k^* \mathbf{D}_\ell^* (\mathbf{xy}^* + \mathbf{yx}^*) \mathbf{D}_\ell \mathbf{f}_k$ is real valued and so we can just as well use squares. For exposition purposes, set

$$\mathbf{A}_k(\mathbf{D}) = |\mathbf{f}_k^* \mathbf{D}^* \mathbf{x}|^2 \mathbf{f}_k \mathbf{f}_k^*, \quad \mathbf{B}_k(\mathbf{D}) = (\mathbf{f}_k^* \mathbf{D}^* \mathbf{x})^2 \mathbf{f}_k \mathbf{f}_k^T.$$

A simple computation we omit yields

$$\begin{aligned} \left(\mathbf{f}_k^* \mathbf{D}^* \mathbf{x} \mathbf{y}^* \mathbf{D} \mathbf{f}_k + \mathbf{f}_k^* \mathbf{D}^* \mathbf{y} \mathbf{x}^* \mathbf{D} \mathbf{f}_k \right)^2 &= \begin{bmatrix} \mathbf{y} \\ \bar{\mathbf{y}} \end{bmatrix}^* \begin{bmatrix} \mathbf{D} & 0 \\ 0 & \mathbf{D}^* \end{bmatrix} \begin{bmatrix} \mathbf{A}_k(\mathbf{D}) & \mathbf{B}_k(\mathbf{D}) \\ \overline{\mathbf{B}_k(\mathbf{D})} & \overline{\mathbf{A}_k(\mathbf{D})} \end{bmatrix} \begin{bmatrix} \mathbf{D}^* & 0 \\ 0 & \mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \bar{\mathbf{y}} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{y} \\ \bar{\mathbf{y}} \end{bmatrix}^* \mathbf{W}_k(\mathbf{D}) \begin{bmatrix} \mathbf{y} \\ \bar{\mathbf{y}} \end{bmatrix}, \end{aligned}$$

where

$$\mathbf{W}_k(\mathbf{D}) := \begin{bmatrix} \mathbf{D} & 0 \\ 0 & \mathbf{D}^* \end{bmatrix} \begin{bmatrix} \mathbf{A}_k(\mathbf{D}) & \mathbf{B}_k(\mathbf{D}) \\ \overline{\mathbf{B}_k(\mathbf{D})} & \overline{\mathbf{A}_k(\mathbf{D})} \end{bmatrix} \begin{bmatrix} \mathbf{D}^* & 0 \\ 0 & \mathbf{D} \end{bmatrix}.$$

Fix a positive threshold T_n . We now claim that (15.1.10) follows from

$$\frac{1}{nL} \sum_{\ell=1}^L \sum_{k=1}^n \mathbf{W}_k(\mathbf{D}_\ell) \mathbf{1}(|\mathbf{f}_k^* \mathbf{D}_\ell^* \mathbf{x}| \leq T_n) \geq \alpha \begin{bmatrix} \mathbf{x} \\ -\bar{\mathbf{x}} \end{bmatrix}^* \begin{bmatrix} \mathbf{x} \\ -\bar{\mathbf{x}} \end{bmatrix} + (1-\delta)^2 \mathbf{I}_{2n} \quad (15.1.11)$$

in which α is any real valued number. To see why this is true, observe that

$$\begin{aligned} \frac{1}{nL} \|\mathcal{A}(\mathbf{xy}^* + \mathbf{yx}^*)\|_{\ell_2}^2 &\geq \begin{bmatrix} \mathbf{y} \\ \bar{\mathbf{y}} \end{bmatrix}^* \frac{1}{nL} \sum_{\ell} \sum_k \mathbf{W}_k(\mathbf{D}_\ell) \mathbf{1}(|\mathbf{f}_k^* \mathbf{D}_\ell^* \mathbf{x}| \leq T_n) \begin{bmatrix} \mathbf{y} \\ \bar{\mathbf{y}} \end{bmatrix} \\ &\geq (1-\delta)^2 \begin{bmatrix} \mathbf{y} \\ \bar{\mathbf{y}} \end{bmatrix}^* \mathbf{I}_{2n} \begin{bmatrix} \mathbf{y} \\ \bar{\mathbf{y}} \end{bmatrix} = 2(1-\delta)^2 \|\mathbf{y}\|_{\ell_2}^2. \end{aligned}$$

The last inequality comes from (15.1.11) together with

$$\begin{bmatrix} \mathbf{x} \\ -\bar{\mathbf{x}} \end{bmatrix}^* \begin{bmatrix} \mathbf{y} \\ \bar{\mathbf{y}} \end{bmatrix} = 0,$$

which holds since we assumed that $\mathbf{x}^* \mathbf{y}$ is real valued.

The remainder of the proof justifies (15.1.11) by means of the matrix Hoeffding inequality. Let $\langle \mathbf{W} \rangle$ be the left-hand side in (15.1.11) (we use notation from physics to denote empirical averages since a bar denotes complex conjugation and we would like to avoid overloading symbols). By definition $\langle \mathbf{W} \rangle$ is the empirical average of L i.i.d. copies of

$$\mathbf{W}(\mathbf{D}) = \frac{1}{n} \sum_{k=1}^n \mathbf{W}_k(\mathbf{D}) \mathbb{1}_{\{|f_k^* \mathbf{D}^* \mathbf{x}| \leq T_n\}}.$$

First, $\mathbf{W}_k(\mathbf{D}) \succeq \mathbf{0}$ since

$$\begin{bmatrix} \mathbf{A}_k(\mathbf{D}) & \mathbf{B}_k(\mathbf{D}) \\ \overline{\mathbf{B}_k(\mathbf{D})} & \overline{\mathbf{A}_k(\mathbf{D})} \end{bmatrix} = \begin{bmatrix} (\mathbf{f}_k^* \mathbf{D}^* \mathbf{x}) \mathbf{f}_k \\ (\overline{\mathbf{f}_k^* \mathbf{D}^* \mathbf{x}}) \overline{\mathbf{f}_k} \end{bmatrix} \begin{bmatrix} (\mathbf{f}_k^* \mathbf{D}^* \mathbf{x}) \mathbf{f}_k \\ (\overline{\mathbf{f}_k^* \mathbf{D}^* \mathbf{x}}) \overline{\mathbf{f}_k} \end{bmatrix}^*.$$

Further,

$$\begin{aligned} \begin{bmatrix} \mathbf{A}_k(\mathbf{D}) & \mathbf{B}_k(\mathbf{D}) \\ \overline{\mathbf{B}_k(\mathbf{D})} & \overline{\mathbf{A}_k(\mathbf{D})} \end{bmatrix} &= \begin{bmatrix} 2\mathbf{A}_k(\mathbf{D}) & \mathbf{0} \\ \mathbf{0} & 2\overline{\mathbf{A}_k(\mathbf{D})} \end{bmatrix} - \begin{bmatrix} \mathbf{A}_k(\mathbf{D}) & -\mathbf{B}_k(\mathbf{D}) \\ -\overline{\mathbf{B}_k(\mathbf{D})} & \overline{\mathbf{A}_k(\mathbf{D})} \end{bmatrix} \\ &\leq \begin{bmatrix} 2\mathbf{A}_k(\mathbf{D}) & \mathbf{0} \\ \mathbf{0} & 2\overline{\mathbf{A}_k(\mathbf{D})} \end{bmatrix}. \end{aligned}$$

The inequality comes from

$$\begin{bmatrix} \mathbf{A}_k(\mathbf{D}) & -\mathbf{B}_k(\mathbf{D}) \\ -\overline{\mathbf{B}_k(\mathbf{D})} & \overline{\mathbf{A}_k(\mathbf{D})} \end{bmatrix} = \begin{bmatrix} (\mathbf{f}_k^* \mathbf{D}^* \mathbf{x}) \mathbf{f}_k \\ -(\overline{\mathbf{f}_k^* \mathbf{D}^* \mathbf{x}}) \overline{\mathbf{f}_k} \end{bmatrix} \begin{bmatrix} (\mathbf{f}_k^* \mathbf{D}^* \mathbf{x}) \mathbf{f}_k \\ -(\overline{\mathbf{f}_k^* \mathbf{D}^* \mathbf{x}}) \overline{\mathbf{f}_k} \end{bmatrix}^* \succeq \mathbf{0}.$$

Hence,

$$\begin{aligned} \sum_{k=1}^n \begin{bmatrix} \mathbf{A}_k(\mathbf{D}) & \mathbf{B}_k(\mathbf{D}) \\ \overline{\mathbf{B}_k(\mathbf{D})} & \overline{\mathbf{A}_k(\mathbf{D})} \end{bmatrix} \mathbb{1}_{\{|f_k^* \mathbf{D}^* \mathbf{x}| \leq T_n\}} &\leq 2 \sum_{k=1}^n \begin{bmatrix} |f_k^* \mathbf{D}^* \mathbf{x}|^2 \mathbf{f}_k \mathbf{f}_k^* & \mathbf{0} \\ \mathbf{0} & |f_k^* \mathbf{D}^* \mathbf{x}|^2 \overline{\mathbf{f}_k} \mathbf{f}_k^T \end{bmatrix} \mathbb{1}_{\{|f_k^* \mathbf{D}^* \mathbf{x}| \leq T_n\}} \\ &\leq 2T_n^2 \sum_{k=1}^n \begin{bmatrix} \mathbf{f}_k \mathbf{f}_k^* & \mathbf{0} \\ \mathbf{0} & \overline{\mathbf{f}_k} \mathbf{f}_k^T \end{bmatrix} \\ &= 2nT_n^2 \mathbf{I}_{2n}. \end{aligned}$$

In summary,

$$\|\mathbf{W}(\mathbf{D})\| \leq 2T_n^2 \|\mathbf{D}\|^2 \leq 2M^2 T_n^2.$$

We now roughly estimate the mean of $\mathbf{W}(\mathbf{D})$. Obviously,

$$\begin{aligned} \mathbf{W}(\mathbf{D}) &= \frac{1}{n} \sum_k \mathbf{W}_k(\mathbf{D}) - \frac{1}{n} \sum_k \mathbf{W}_k(\mathbf{D}) \mathbb{1}_{\{|\mathbf{f}_k^* \mathbf{D}^* \mathbf{x}| \leq T_n\}} \\ &:= \tilde{\mathbf{W}}(\mathbf{D}) - \frac{1}{n} \sum_k \mathbf{W}_k(\mathbf{D}) \mathbb{1}_{\{|\mathbf{f}_k^* \mathbf{D}^* \mathbf{x}| \leq T_n\}}. \end{aligned}$$

By Lemmas 15.1.1 and 15.1.2, the mean of the first term ($\tilde{\mathbf{W}}(\mathbf{D})$) is equal to

$$\mathbb{E} \tilde{\mathbf{W}}(\mathbf{D}) = \mathbf{I}_{2n} + \begin{bmatrix} \mathbf{x} \mathbf{x}^* & 2\mathbf{x} \mathbf{x}^T \\ 2\bar{\mathbf{x}} \mathbf{x}^* & \bar{\mathbf{x}} \mathbf{x}^T \end{bmatrix}. \quad (15.1.12)$$

Furthermore, a simple calculation shows that since

$$|\mathbf{f}_k^* \mathbf{D}^* \mathbf{x}| \leq \|\mathbf{f}_k\|_{\ell_2} \|\mathbf{D}^* \mathbf{x}\|_{\ell_2} \leq \|\mathbf{f}_k\|_{\ell_2} \|\mathbf{D}\| \leq \sqrt{n} \|\mathbf{D}\|,$$

one can verify that

$$\|\mathbf{W}_k(\mathbf{D})\| \leq 4n^2 \|\mathbf{D}\|^4 \leq 4M^4 n^2.$$

Therefore, Jensen's inequality gives

$$\begin{aligned} \left\| \mathbb{E} \mathbf{W}(\mathbf{D}) - \mathbb{E} \tilde{\mathbf{W}}(\mathbf{D}) \right\| &= \left\| \mathbb{E} \frac{1}{n} \sum_k \mathbf{W}_k(\mathbf{D}) \mathbb{1}_{\{|\mathbf{f}_k^* \mathbf{D}^* \mathbf{x}| \leq T_n\}} \right\| \\ &\leq \mathbb{E} \left\| \frac{1}{n} \sum_k \mathbf{W}_k(\mathbf{D}) \mathbb{1}_{\{|\mathbf{f}_k^* \mathbf{D}^* \mathbf{x}| \leq T_n\}} \right\| \\ &\leq 4M^4 n \sum_{k=1}^n \mathbb{P}(|\mathbf{f}_k^* \mathbf{D}^* \mathbf{x}| > T_n). \end{aligned}$$

Setting $T_n = \sqrt{2\beta \log n}$, then a simple application of Hoeffding's inequality gives

$$\mathbb{P}(|\mathbf{f}_k^* \mathbf{D}^* \mathbf{x}| > \sqrt{2\beta \log n}) \leq 2n^{-\beta}$$

(we omit the details). Therefore,

$$\|\mathbb{E} \mathbf{W}(\mathbf{D}) - \mathbb{E} \tilde{\mathbf{W}}(\mathbf{D})\| \leq \frac{8M^4}{n^{\beta-2}}. \quad (15.1.13)$$

Next,

$$\|\mathbf{W}(\mathbf{D}) - \mathbb{E} \mathbf{W}(\mathbf{D})\| \leq \|\mathbf{W}(\mathbf{D})\| + \|\mathbb{E} \mathbf{W}(\mathbf{D})\| \leq \|\mathbf{W}(\mathbf{D})\| + \|\mathbb{E} \tilde{\mathbf{W}}(\mathbf{D})\| + \|\mathbb{E} \mathbf{W}(\mathbf{D}) - \mathbb{E} \tilde{\mathbf{W}}(\mathbf{D})\|$$

and with T_n as above, collecting our estimates gives

$$\|\mathbf{W}(\mathbf{D}) - \mathbb{E} \mathbf{W}(\mathbf{D})\| \leq 4M^2\beta \log n + 4 + \frac{8M^4}{n^{\beta-2}} := \Delta.$$

We have done the groundwork to apply the matrix Hoeffding inequality (15.1.3), which reads

$$\mathbb{P}(\|\langle \mathbf{W} \rangle - \mathbb{E} \mathbf{W}(\mathbf{D})\| \geq t) \leq 2n \exp\left(-\frac{Lt^2}{8\Delta^2}\right).$$

This implies that when β is sufficiently large and $L \geq c \log^3 n$ for a sufficiently large constant,

$$\|\langle \mathbf{W} \rangle - \mathbb{E} \mathbf{W}(\mathbf{D})\| \leq \epsilon/2$$

with probability at least $1 - 1/(2n)$. Now from (15.1.12), and (15.1.13) gives

$$\mathbb{E} \mathbf{W}(\mathbf{D}) = \mathbf{I}_{2n} + \frac{3}{2} \begin{bmatrix} \mathbf{x} \\ \bar{\mathbf{x}} \end{bmatrix} [\mathbf{x}^*, \mathbf{x}^T] - \frac{1}{2} \begin{bmatrix} \mathbf{x} \\ -\bar{\mathbf{x}} \end{bmatrix} [\mathbf{x}^*, -\mathbf{x}^T] + \mathbf{E},$$

where (15.1.13) gives that $\|\mathbf{E}\| \leq \epsilon/2$ provided $\beta \geq 2 + \log(16M^4\epsilon^{-1})/\log n$. Hence, we have established that

$$\langle \mathbf{W} \rangle \succeq (1 - \epsilon) \mathbf{I}_{2n} + \frac{3}{2} \begin{bmatrix} \mathbf{x} \\ \bar{\mathbf{x}} \end{bmatrix} [\mathbf{x}^*, \mathbf{x}^T] - \frac{1}{2} \begin{bmatrix} \mathbf{x} \\ -\bar{\mathbf{x}} \end{bmatrix} [\mathbf{x}^*, -\mathbf{x}^T] \succeq (1 - \epsilon) \mathbf{I}_{2n} - \frac{1}{2} \begin{bmatrix} \mathbf{x} \\ -\bar{\mathbf{x}} \end{bmatrix} [\mathbf{x}^*, -\mathbf{x}^T]$$

since $\begin{bmatrix} \mathbf{x} \\ \bar{\mathbf{x}} \end{bmatrix} [\mathbf{x}^*, \mathbf{x}^T] \succeq \mathbf{0}$. With $\epsilon = 2\delta - \delta^2$, this is the desired conclusion (15.1.11). ■

15.1.4 Dual certificate construction via the golfing scheme

We now construct the approximate dual certificate \mathbf{Z} obeying the conditions of Lemma 15.1.6. For this purpose we use the golfing scheme first presented in the work of Gross [119]. Modifications of this technique have subsequently been used in many other papers e.g. [56, 64, 148]. The special form used here is most closely related to the construction in [148]. The mathematical validity of our construction crucially relies on the lemma below, whose proof is the object of the separate Section 15.1.5.

Lemma 15.1.8 *Assume that $L \geq c \log^3 n$ for a sufficiently large constant c . Then for any fixed $\mathbf{X} \in T$, there exists \mathbf{Y} of the form $\mathbf{Y} = \mathcal{A}^*(\boldsymbol{\lambda})$ with $\boldsymbol{\lambda}$ real valued such that*

$$\|\mathbf{Y} - \mathbf{X}\| \leq \frac{\sqrt{2}}{20} \|\mathbf{X}\|_F$$

holds with probability at least $1 - 1/n^2$. This inequality has the immediate consequences

$$\|\mathbf{Y}_T - \mathbf{X}\|_F \leq \frac{1}{5} \|\mathbf{X}\|_F, \quad \|\mathbf{Y}_{T^\perp}\| \leq \frac{\sqrt{2}}{20} \|\mathbf{X}\|_F.$$

To build our approximate dual certificate \mathbf{Z} , we partition the modulations or CDPs into $B + 1$ different groups so that, from now on, \mathcal{A}_0 corresponds to those measurements from the first L_0 modulations, \mathcal{A}_1 to those from the next L_1 ones, and so on. Clearly, $L_0 + L_1 + \dots + L_B = L$. The random mappings $\{\mathcal{A}_b\}_{b=0}^B$ correspond to independent modulations and are thus independent. Our golfing scheme starts with $\mathbf{X}^{(0)} = \frac{2}{nL_0} \mathcal{P}_T(\mathcal{A}_0^*(\mathbf{1}))$ ($\mathbf{1}$ is the all-one vector) and for $b = 1, \dots, B$, inductively defines

- $\mathbf{Y}^{(b)} \in \text{Range}(\mathcal{A}_b^*)$ obeying $\|\mathbf{Y}^{(b)} - \mathbf{X}^{(b-1)}\| \leq \frac{\sqrt{2}}{20} \|\mathbf{X}^{(b-1)}\|_F$,
- and $\mathbf{X}^{(b)} = \mathbf{X}^{(b-1)} - \mathcal{P}_T(\mathbf{Y}^{(b)})$.

In the end, we set

$$\mathbf{Z} = \mathbf{Y} - \frac{2}{nL_0} \mathcal{A}_0^*(\mathbf{1}), \quad \mathbf{Y} = \sum_{t=1}^B \mathbf{Y}^{(t)}.$$

Note that Lemma 15.1.8 asserts that $\mathbf{Y}^{(b)}$ exists with high probability, and that for each b both

$$\|\mathbf{X}^{(b)}\|_F \leq \frac{1}{5} \|\mathbf{X}^{(b-1)}\|_F \quad \text{and} \quad \|\mathbf{Y}_{T^\perp}^{(b)}\| \leq \frac{\sqrt{2}}{20} \|\mathbf{X}^{(b-1)}\|_F \quad (15.1.14)$$

hold on an event of probability at least $1 - 1/n^2$.

We now show that our construction \mathbf{Z} satisfies the required assumptions from Lemma 15.1.6. First, \mathbf{Z} is self-adjoint and of the form $\mathcal{A}^*(\boldsymbol{\lambda})$ with $\boldsymbol{\lambda} \in \mathbb{R}^{nL}$. Second,

$$\mathbf{Z}_T = \mathbf{Y}_T - \frac{2}{nL_0} \mathcal{P}_T(\mathcal{A}_0^*(\mathbf{1})) = \sum_{b=1}^B \mathcal{P}_T(\mathbf{Y}^{(b)}) - \mathbf{X}^{(0)} = \sum_{b=1}^B (\mathbf{X}^{(b-1)} - \mathbf{X}^{(b)}) - \mathbf{X}^{(0)} = -\mathbf{X}^{(B)}.$$

Then (15.1.14) implies that with probability at least $1 - B/n^2$

$$\|\mathbf{Z}_T\|_F \leq \|\mathbf{X}^{(B)}\|_F \leq \frac{1}{5^B} \|\mathbf{X}^{(0)}\|_F. \quad (15.1.15)$$

Also, (15.1.14) gives

$$\|\mathbf{Y}_{T^\perp}\| \leq \sum_{b=1}^B \|\mathbf{Y}_{T^\perp}^{(b)}\| \leq \frac{\sqrt{2}}{20} \sum_{b=1}^B \|\mathbf{X}^{(t-1)}\|_F \leq \frac{\sqrt{2}}{20} \sum_{b=1}^B \frac{1}{5^b} \|\mathbf{X}^{(0)}\|_F < \frac{\sqrt{2}}{16} \|\mathbf{X}^{(0)}\|_F \quad (15.1.16)$$

with probability at least $1 - B/n^2$. If $L_0 \geq c \log n$ for a sufficiently large constant $c > 0$, Lemma 15.1.3 states that

$$\left\| \frac{2}{nL_0} \mathcal{A}_0^*(\mathbf{1}) - 2\mathbf{I} \right\| \leq \frac{1}{4}$$

with probability at least $1 - 1/n^2$. Using the fact that for any matrix \mathbf{W} , we have $\|\mathbf{W}_T\| \leq 2\|\mathbf{W}\|$ and $\|\mathbf{W}_{T^\perp}\| \leq \|\mathbf{W}\|$ we conclude that

$$\|\mathbf{X}^{(0)} - 2\mathbf{I}_T\| \leq 1/2, \quad \|\mathbf{Y}_{T^\perp} - \mathbf{Z}_{T^\perp} - 2\mathbf{I}_{T^\perp}\| \leq 1/4. \quad (15.1.17)$$

Since $\mathbf{X}^{(0)}$ has rank at most 2,

$$\|\mathbf{X}^{(0)}\|_F \leq \sqrt{2} \|\mathbf{X}^{(0)}\| \leq \sqrt{2} \|\mathbf{X}^{(0)} - 2\mathbf{I}_T\| + 2\sqrt{2} \|\mathbf{I}_T\|$$

Finally, with (15.1.17) and $\|\mathbf{I}_T\| \leq 1$, we conclude that

$$\|\mathbf{X}^{(0)}\|_F < 4. \quad (15.1.18)$$

Plugging this into (15.1.15) we arrive at

$$\|\mathbf{Z}_T\|_F \leq \frac{4}{5^B}. \quad (15.1.19)$$

Also, (15.1.16), (15.1.17) and (15.1.18) give

$$\|\mathbf{Z}_{T^\perp} + 2\mathbf{I}_{T^\perp}\| \leq \|\mathbf{Y}_{T^\perp}\| + \|\mathbf{Y}_{T^\perp} - \mathbf{Z}_{T^\perp} - 2\mathbf{I}_{T^\perp}\| \leq \frac{\sqrt{2}}{4} + \frac{1}{4} < 1 \quad \Rightarrow \quad \mathbf{Z}_{T^\perp} \leq -\mathbf{I}_{T^\perp}. \quad (15.1.20)$$

Therefore, the assumptions in Lemma 15.1.6 hold with probability at least $1 - 1/2n$ by applying the union bound and using with the proviso that $B \geq c_1 \log n$ and $L_b \geq c_2 \log^3 n$ for sufficiently large constants c_1 and c_2 (this is why we require $L \geq c \log^4 n$ for a sufficiently large constant).

15.1.5 Proof of Lemma 15.1.8

The immediate consequences hold for the following reasons. First, since any matrix in T has rank at most 2,

$$\|\mathbf{Y}_T - \mathbf{X}\|_F \leq \sqrt{2}\|\mathbf{Y}_T - \mathbf{X}\| \leq 2\sqrt{2}\|\mathbf{Y} - \mathbf{X}\| \leq \frac{1}{5}\|\mathbf{X}\|_F,$$

where the second inequality follows from $\|\mathbf{M}_T\| \leq 2\|\mathbf{M}\|$ for any \mathbf{M} . Second, since $\|\mathbf{M}_{T^\perp}\| \leq \|\mathbf{M}\|$,

$$\|\mathbf{Y}_{T^\perp}\| = \|\mathbf{Y}_{T^\perp} - \mathbf{X}_{T^\perp}\| \leq \|\mathbf{Y} - \mathbf{X}\| \leq \frac{\sqrt{2}}{20}\|\mathbf{X}\|_F.$$

It thus suffices to prove the first property. To this end consider the eigenvalue decomposition of $\mathbf{X} = \lambda_1 \mathbf{u}_1 \mathbf{u}_1^* + \lambda_2 \mathbf{u}_2 \mathbf{u}_2^*$. The proof follows from Lemma 15.1.9 below combined with Lemma 15.1.3.

Lemma 15.1.9 Assume $L \geq c \log^3 n$ for a sufficiently large constant c . Given any fixed self-adjoint matrix $\mathbf{v}\mathbf{v}^*$, with probability at least $1 - 1/(2n^3)$ there exists $\tilde{\mathbf{Y}} \in \text{Range}(\mathcal{A}^*)$ obeying

$$\|\tilde{\mathbf{Y}} - (\mathbf{v}\mathbf{v}^* + \|\mathbf{v}\|_{\ell_2}^2 \mathbf{I})\| \leq \epsilon \|\mathbf{v}\|_{\ell_2}^2.$$

Proof Without loss of generality, assume $\|\mathbf{v}\|_{\ell_2} = 1$ and set $\tilde{\mathbf{Y}} = \langle \mathbf{Y} \rangle$,

$$\langle \mathbf{Y} \rangle = \frac{1}{L} \sum_{l=1}^L \mathbf{Y}_l, \quad \mathbf{Y}_l = \frac{1}{n} \sum_{k=1}^n |\mathbf{f}_k^* \mathbf{D}_l^* \mathbf{v}|^2 \mathbb{1}_{\{|\mathbf{f}_k^* \mathbf{D}_l^* \mathbf{v}| \leq T_n\}} \mathbf{D}_l \mathbf{f}_k \mathbf{f}_k^* \mathbf{D}_l^*,$$

which is of the form $\mathcal{A}^*(\boldsymbol{\lambda})$. The \mathbf{Y}_l 's are i.i.d. copies of \mathbf{Y} ,

$$\mathbf{Y} = \frac{1}{n} \sum_{k=1}^n |\mathbf{f}_k^* \mathbf{D}^* \mathbf{v}|^2 \mathbf{D} \mathbf{f}_k \mathbf{f}_k^* \mathbf{D}^* - \frac{1}{n} \sum_{k=1}^n |\mathbf{f}_k^* \mathbf{D}^* \mathbf{v}|^2 \mathbb{1}_{\{|\mathbf{f}_k^* \mathbf{D}^* \mathbf{v}| > T_n\}} \mathbf{D} \mathbf{f}_k \mathbf{f}_k^* \mathbf{D}^*.$$

Notice that the random positive semi-definite matrix \mathbf{Y} obeys

$$\mathbf{Y} \leq \frac{1}{n} \sum_{k=1}^n T_n^2 \mathbf{D} \mathbf{f}_k \mathbf{f}_k^* \mathbf{D}^* = T_n^2 \mathbf{D} \mathbf{D}^*.$$

By Lemma 15.1.1,

$$\mathbb{E} \left(\frac{1}{n} \sum_{k=1}^n |\mathbf{f}_k^* \mathbf{D}^* \mathbf{v}|^2 \mathbf{D} \mathbf{f}_k \mathbf{f}_k^* \mathbf{D}^* \right) = \mathbf{v}\mathbf{v}^* + \mathbf{I}.$$

Using Jensen's inequality, we have as in the proof of Lemma 15.1.7

$$\left\| \mathbb{E} \left(\frac{1}{n} \sum_{k=1}^n |\mathbf{f}_k^* \mathbf{D}^* \mathbf{v}|^2 \mathbb{1}_{\{|\mathbf{f}_k^* \mathbf{D}^* \mathbf{v}| > T_n\}} \mathbf{D} \mathbf{f}_k \mathbf{f}_k^* \mathbf{D}^* \right) \right\| \leq \frac{1}{n} \sum_{k=1}^n \mathbb{E}(\mathbb{1}_{\{|\mathbf{f}_k^* \mathbf{D}^* \mathbf{v}| > T_n\}}) n^2 M^4. \quad (15.1.21)$$

Put $T_n = \sqrt{2\beta \log n}$. Hoeffding's inequality gives

$$\mathbb{E}(\mathbb{1}_{\{|\mathbf{f}_k^* \mathbf{D}^* \mathbf{v}| > T_n\}}) \leq 2n^{-\beta}.$$

Plugging this into (15.1.21) we arrive at

$$\left\| \mathbb{E} \left(\frac{1}{n} \sum_{k=1}^n |\mathbf{f}_k^* \mathbf{D}^* \mathbf{v}|^2 \mathbb{1}_{\{|\mathbf{f}_k^* \mathbf{D}^* \mathbf{v}| > T_n\}} \mathbf{D} \mathbf{f}_k \mathbf{f}_k^* \mathbf{D}^* \right) \right\| \leq \frac{2M^4}{n^{\beta-2}}.$$

For sufficiently large β , this implies

$$\|\mathbb{E}(\mathbf{Y}) - (\mathbf{v}\mathbf{v}^* + \mathbf{I})\| \leq \frac{2\|\mathbf{D}\|^4}{n^{\beta-2}} \leq \frac{\epsilon}{2}. \quad (15.1.22)$$

By using Hoeffding inequality in a similar fashion as in the proof of Lemma 15.1.7, we obtain (we omit the details)

$$\|\langle \mathbf{Y} \rangle - \mathbb{E}(\mathbf{Y})\| \leq \frac{\epsilon}{2}.$$

Combining the latter with (15.1.22), we conclude

$$\|\langle \mathbf{Y} \rangle - (\mathbf{v}\mathbf{v}^* + \mathbf{I})\| \leq \epsilon.$$

■

15.2 Proof of stability of PhaseLift with CDP measurements

We begin the stability proofs by introducing some notation. We define

$$\|\mathcal{A}\|_{1-1} = \max_{\mathbf{X} \geq \mathbf{0}} \frac{\|\mathcal{A}(\mathbf{X})\|_{\ell_1}}{\text{tr}(\mathbf{X})}.$$

We remind the readers that

$$T = \{\mathbf{x}\mathbf{y}^* + \mathbf{y}\mathbf{x}^*\}.$$

We use \mathcal{A}_T^{-1} to denote the inversion of \mathcal{A} restricted to T . To establish our result we prove the following intermediate lemma.

Lemma 15.2.1 *Suppose that the mapping \mathcal{A} obeys the following properties for all*

matrices $\mathbf{X} \in T$

$$\frac{1}{\sqrt{nL}} \|\mathcal{A}(\mathbf{X})\|_{\ell_2} \geq \frac{(1-\delta)}{\sqrt{2}} \|\mathbf{X}\|_F \quad \text{and} \quad \|\mathcal{A}\|_{1-1} \leq M^2 n L. \quad (15.2.1)$$

Furthermore, there exists a self-adjoint matrix of the form $\mathbf{Z} = \mathcal{A}^*(\boldsymbol{\lambda})$ (λ is a real vector) obeying

$$\mathbf{Z}_{T^\perp} \preceq -\mathbf{I}_{T^\perp}, \quad \|\mathbf{Z}_T\|_F \leq \frac{1-\delta}{2M^2\sqrt{nL}} \quad \text{and} \quad \|\boldsymbol{\lambda}\|_{\ell_2} \leq \frac{c}{\sqrt{nL}}, \quad (15.2.2)$$

for c a fixed numerical constant. Then

$$\|\hat{\mathbf{X}} - \mathbf{x}\mathbf{x}^*\|_F \leq C \|\mathbf{w}\|_{\ell_2},$$

where C is also a fixed numerical constant.

Proof Define $\mathbf{H} = \hat{\mathbf{X}} - \mathbf{x}\mathbf{x}^*$ and note that by the optimality of $\hat{\mathbf{X}}$ for (11.4.1) we have

$$\|\mathcal{A}(\mathbf{H})\|_{\ell_2} = \|\mathcal{A}(\hat{\mathbf{X}}) - \mathbf{y}\|_{\ell_2} \leq \|\mathcal{A}(\mathbf{x}\mathbf{x}^*) - \mathbf{y}\|_{\ell_2} = \|\mathbf{w}\|_{\ell_2}.$$

Next, note that

$$\begin{aligned} \|\mathbf{H}_T\|_F &\leq \|\mathcal{A}_T^{-1} \mathcal{A}(\mathbf{H}_T)\|_F \\ &\leq \|\mathcal{A}_T^{-1}\| \|\mathcal{A}(\mathbf{H}_T)\|_{\ell_2} \\ &\leq \|\mathcal{A}_T^{-1}\| (\|\mathcal{A}(\mathbf{H})\|_{\ell_2} + \|\mathcal{A}(\mathbf{H}_{T^\perp})\|_{\ell_2}) \\ &\leq \|\mathcal{A}_T^{-1}\| (\|\mathcal{A}(\mathbf{H})\|_{\ell_2} + \|\mathcal{A}(\mathbf{H}_{T^\perp})\|_{\ell_1}) \\ &\leq \|\mathcal{A}_T^{-1}\| \|\mathbf{w}\|_{\ell_2} + \|\mathcal{A}_T^{-1}\| \|\mathcal{A}\|_{1-1} \text{tr}(\mathbf{H}_{T^\perp}) \\ &\leq \frac{\sqrt{2}}{\sqrt{m}(1-\delta)} \|\mathbf{w}\|_{\ell_2} + \frac{\sqrt{2}}{\sqrt{m}(1-\delta)} \|\mathcal{A}\|_{1-1} \text{tr}(\mathbf{H}_{T^\perp}) \\ &\leq \frac{\sqrt{2}}{\sqrt{nL}(1-\delta)} \|\mathbf{w}\|_{\ell_2} + M^2 \frac{\sqrt{2nL}}{(1-\delta)} \text{tr}(\mathbf{H}_{T^\perp}) \end{aligned}$$

We also have

$$\begin{aligned}
|\langle \mathbf{H}, \mathbf{Z} \rangle| &= |\langle \mathbf{H}, \mathcal{A}^*(\boldsymbol{\lambda}) \rangle| \\
&\leq |\langle \mathcal{A}(\mathbf{H}), \boldsymbol{\lambda} \rangle| \\
&\leq \|\mathcal{A}(\mathbf{H})\|_{\ell_2} \|\boldsymbol{\lambda}\|_{\ell_2} \\
&\leq c \frac{\|\mathbf{w}\|_{\ell_2}}{\sqrt{nL}}.
\end{aligned} \tag{15.2.3}$$

On the one hand using (15.2.3),

$$\begin{aligned}
\langle \mathbf{H}_T, \mathbf{Z}_T \rangle &= \langle \mathbf{H}, \mathbf{Z} \rangle - \langle \mathbf{H}_{T^\perp}, \mathbf{Z}_{T^\perp} \rangle \\
&\geq \langle \mathbf{H}, \mathbf{Z} \rangle + \langle \mathbf{H}_{T^\perp}, \mathbf{I}_{T^\perp} \rangle \\
&= \langle \mathbf{H}, \mathbf{Z} \rangle + \text{trace}(\mathbf{H}_{T^\perp}) \\
&\geq -c \frac{\|\mathbf{w}\|_{\ell_2}}{\sqrt{nL}} + \text{trace}(\mathbf{H}_{T^\perp})
\end{aligned} \tag{15.2.4}$$

Therefore,

$$\text{trace}(\mathbf{H}_{T^\perp}) \geq \frac{1}{M^2 n L} \|\mathcal{A}(\mathbf{H}_{T^\perp})\|_{\ell_1} \geq \frac{1}{M^2 n L} \|\mathcal{A}(\mathbf{H}_{T^\perp})\|_{\ell_2}.$$

where the first inequality above follows from the second condition in (15.2.1). The injectivity property (first condition of (15.2.1)) gives

$$\frac{1}{\sqrt{nL}} \|\mathcal{A}(\mathbf{H}_T)\|_{\ell_2} \geq \frac{(1-\delta)}{\sqrt{2}} \|\mathbf{H}_T\|_F$$

and since $-\mathcal{A}(\mathbf{H}_{T^\perp}) = \mathcal{A}(\mathbf{H}_T) - \mathcal{A}(\mathbf{H})$, we established

$$\begin{aligned}
\text{trace}(\mathbf{H}_{T^\perp}) &\geq \frac{1}{M^2 n L} \|\mathcal{A}(\mathbf{H}_{T^\perp})\|_{\ell_2} \\
&\geq \frac{1}{M^2 n L} (\|\mathcal{A}(\mathbf{H}_T)\|_{\ell_2} - \|\mathcal{A}(\mathbf{H})\|_{\ell_2}) \\
&\geq \frac{1-\delta}{\sqrt{2nL} M^2} \|\mathbf{H}_T\|_F - \frac{1}{M^2 n L} \|\mathbf{w}\|_{\ell_2}.
\end{aligned} \tag{15.2.5}$$

On the other hand,

$$|\langle \mathbf{H}_T, \mathbf{Z}_T \rangle| \leq \|\mathbf{H}_T\|_F \|\mathbf{Z}_T\|_F \leq \frac{1-\delta}{2M^2\sqrt{nL}} \|\mathbf{H}_T\|_F. \quad (15.2.6)$$

In summary, (15.2.4), (15.2.5) and (15.2.6) assert that

$$\left(1 - \frac{1}{\sqrt{2}}\right) \frac{1-\delta}{\sqrt{2nLM^2}} \|\mathbf{H}_T\|_F \leq \frac{1}{\sqrt{nL}} \left(c - \frac{1}{M^2\sqrt{nL}}\right) \|\mathbf{w}\|_{\ell_2} \Rightarrow \|\mathbf{H}_T\|_F \leq C' \|\mathbf{w}\|_{\ell_2}. \quad (15.2.7)$$

Note that $\mathbf{H}_{T^\perp} \succeq \mathbf{0}$ so $\text{trace}(\mathbf{H}_{T^\perp}) = \|\mathbf{H}_{T^\perp}\|_* \geq \|\mathbf{H}_{T^\perp}\|_F$. The latter together with (15.2.4) and (15.2.6) give

$$\|\mathbf{H}_{T^\perp}\|_F \leq c \frac{\|\mathbf{w}\|_{\ell_2}}{\sqrt{nL}} + \frac{1-\delta}{2M^2\sqrt{nL}} \|\mathbf{H}_T\|_F. \quad (15.2.8)$$

The result follows from combining (15.2.7) and (15.2.8). ■

15.3 Proofs for Wirtinger flow

15.3.1 Preliminaries

We first note that in the CDP model with admissible CDPs $\|\mathbf{a}_r\|_{\ell_2} \leq \sqrt{6n}$ for all $r = 1, 2, \dots, m$. In the Gaussian model the measurements vectors also obey $\|\mathbf{a}_r\|_{\ell_2} \leq \sqrt{6n}$ for all $r = 1, 2, \dots, m$ with probability at least $1 - me^{-1.5n}$. Throughout the proofs, we assume we are on this event without explicitly mentioning it each time.

Before we begin with the proofs we should mention that we will prove our result using the update

$$\mathbf{z}_{\tau+1} = \mathbf{z}_\tau - \frac{\mu}{\|\mathbf{x}\|_{\ell_2}^2} \nabla f(\mathbf{z}_\tau), \quad (15.3.1)$$

in lieu of the WF update

$$\mathbf{z}_{\tau+1} = \mathbf{z}_\tau - \frac{\mu_{\text{WF}}}{\|\mathbf{z}_0\|_{\ell_2}^2} \nabla f(\mathbf{z}_\tau). \quad (15.3.2)$$

Since $\|\mathbf{z}_0\|_{\ell_2}^2 - \|\mathbf{x}\|_{\ell_2}^2 \leq \frac{1}{64} \|\mathbf{x}\|_{\ell_2}^2$ holds with high probability as proven in Section 15.3.8, we have

$$\|\mathbf{z}_0\|_{\ell_2}^2 \geq \frac{63}{64} \|\mathbf{x}\|_{\ell_2}^2. \quad (15.3.3)$$

Therefore, the results for the update (15.3.1) automatically carry over to the WF update with a simple rescaling of the upper bound on the learning parameter. More precisely, if we prove that the update (15.3.1) converges to a global optimum as long as $\mu \leq \mu_0$, then the convergence of the WF update to a global optimum is guaranteed as long as $\mu_{\text{WF}} \leq \frac{63}{64} \mu_0$. Also, the update in (15.3.1) is invariant to the Euclidean norm of \mathbf{x} . Therefore, without loss of generality we will assume throughout the proofs that $\|\mathbf{x}\|_{\ell_2} = 1$.

We remind the reader that throughout \mathbf{x} is a solution to our quadratic equations, i.e. obeys $y = |\mathbf{A}\mathbf{x}|^2$ and that the sampling vectors are independent from \mathbf{x} . Define

$$P := \{\mathbf{x}e^{i\phi} : \phi \in [0, 2\pi]\}.$$

to be the set of all vectors that differ from the planted solution \mathbf{x} only by a global phase factor. We also introduce the set of all points that are close to P ,

$$E(\epsilon) := \{\mathbf{z} \in \mathbb{C}^n : \text{dist}(\mathbf{z}, P) \leq \epsilon\},$$

Finally for any vector $\mathbf{z} \in \mathbb{C}^n$ we define the phase $\phi(\mathbf{z})$ as

$$\phi(\mathbf{z}) := \arg \min_{\phi \in [0, 2\pi]} \|\mathbf{z} - e^{i\phi} \mathbf{x}\|_{\ell_2},$$

so that

$$\text{dist}(\mathbf{z}, \mathbf{x}) = \|\mathbf{z} - e^{i\phi(\mathbf{z})}\mathbf{x}\|_{\ell_2}.$$

15.3.2 Formulas for the complex gradient and Hessian

We gather some useful gradient and Hessian calculations that will be used repeatedly. Starting with

$$f(\mathbf{z}) = \frac{1}{2m} \sum_{r=1}^m (y_r - \bar{\mathbf{z}}^T (\mathbf{a}_r \mathbf{a}_r^*) \mathbf{z})^2 = \frac{1}{2m} \sum_{r=1}^m (y_r - \mathbf{z}^T (\mathbf{a}_r \mathbf{a}_r^*)^T \bar{\mathbf{z}})^2,$$

we establish

$$\left(\frac{\partial}{\partial \mathbf{z}} f(\mathbf{z}) \right)^T = \frac{1}{m} \sum_{r=1}^m (\mathbf{z}^T (\mathbf{a}_r \mathbf{a}_r^*)^T \bar{\mathbf{z}} - y_r) (\mathbf{a}_r \mathbf{a}_r^*)^T \bar{\mathbf{z}}.$$

This gives

$$\nabla f(\mathbf{z}) = \left(\frac{\partial}{\partial \mathbf{z}} f(\mathbf{z}) \right)^* = \frac{1}{m} \sum_{r=1}^m (\bar{\mathbf{z}}^T (\mathbf{a}_r \mathbf{a}_r^*) \mathbf{z} - y_r) (\mathbf{a}_r \mathbf{a}_r^*) \mathbf{z}.$$

For the second derivative, we write

$$\mathcal{H}_{\mathbf{z}\mathbf{z}} = \frac{\partial}{\partial \mathbf{z}} \left(\frac{\partial}{\partial \mathbf{z}} f(\mathbf{z}) \right)^* = \frac{1}{m} \sum_{r=1}^m (2|\mathbf{a}_r^* \mathbf{z}|^2 - y_r) \mathbf{a}_r \mathbf{a}_r^*$$

and

$$\mathcal{H}_{\bar{\mathbf{z}}\mathbf{z}} = \frac{\partial}{\partial \bar{\mathbf{z}}} \left(\frac{\partial}{\partial \mathbf{z}} f(\mathbf{z}) \right)^* = \frac{1}{m} \sum_{r=1}^m (\mathbf{a}_r^* \mathbf{z})^2 \mathbf{a}_r \mathbf{a}_r^T.$$

Therefore,

$$\nabla^2 f(\mathbf{z}) = \frac{1}{m} \sum_{r=1}^m \begin{bmatrix} (2|\mathbf{a}_r^* \mathbf{z}|^2 - y_r) \mathbf{a}_r \mathbf{a}_r^* & (\mathbf{a}_r^* \mathbf{z})^2 \mathbf{a}_r \mathbf{a}_r^T \\ (\overline{\mathbf{a}_r^* \mathbf{z}})^2 \bar{\mathbf{a}}_r \mathbf{a}_r^* & (2|\mathbf{a}_r^* \mathbf{z}|^2 - y_r) \bar{\mathbf{a}}_r \mathbf{a}_r^T \end{bmatrix}.$$

15.3.3 Expectation and concentration

This section gathers some useful intermediate results whose proofs are deferred to Appendix H. The first two lemmas establish the expectation of the Hessian, gradient and a related random variable in both the Gaussian and admissible CDP models.¹

Lemma 15.3.1 *Recall that \mathbf{x} is a solution obeying $\|\mathbf{x}\|_{\ell_2} = 1$, which is independent from the sampling vectors. Furthermore, assume the sampling vectors \mathbf{a}_r are distributed according to either the Gaussian or admissible CDP model. Then*

$$\mathbb{E}[\nabla^2 f(\mathbf{x})] = \mathbf{I}_{2n} + \frac{3}{2} \begin{bmatrix} \mathbf{x} \\ \bar{\mathbf{x}} \end{bmatrix} [\mathbf{x}^*, \mathbf{x}^T] - \frac{1}{2} \begin{bmatrix} \mathbf{x} \\ -\bar{\mathbf{x}} \end{bmatrix} [\mathbf{x}^*, -\mathbf{x}^T].$$

Lemma 15.3.2 *In the setup of Lemma 15.3.1, let $\mathbf{z} \in \mathbb{C}^n$ be a fixed vector independent of the sampling vectors. We have*

$$\mathbb{E}[\nabla f(\mathbf{z})] = (\mathbf{I} - \mathbf{x}\mathbf{x}^*)\mathbf{z} + 2(\|\mathbf{z}\|_{\ell_2}^2 - 1)\mathbf{z}.$$

The next lemma gathers some useful identities in the Gaussian model.

Lemma 15.3.3 *Assume $\mathbf{u}, \mathbf{v} \in \mathbb{C}^n$ are fixed vectors obeying $\|\mathbf{u}\|_{\ell_2} = \|\mathbf{v}\|_{\ell_2} = 1$ which are independent of the sampling vectors. Furthermore, assume the measurement vectors \mathbf{a}_r are distributed according to the Gaussian model. Then*

$$\mathbb{E}[(\operatorname{Re}(\mathbf{u}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{v}))^2] = \frac{1}{2} + \frac{3}{2}(\operatorname{Re}(\mathbf{u}^* \mathbf{v}))^2 - \frac{1}{2}(\operatorname{Im}(\mathbf{u}^* \mathbf{v}))^2 \quad (15.3.4)$$

$$\mathbb{E}[\operatorname{Re}(\mathbf{u}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{v}) |\mathbf{a}_r^* \mathbf{v}|^2] = 2 \operatorname{Re}(\mathbf{u}^* \mathbf{v}) \quad (15.3.5)$$

$$\mathbb{E}[|\mathbf{a}_r^* \mathbf{v}|^{2k}] = k!. \quad (15.3.6)$$

The next lemma establishes the concentration of the Hessian around its mean for both the Gaussian and the CDP model.

Lemma 15.3.4 *In the setup of Lemma 15.3.1, assume the vectors \mathbf{a}_r are distributed according to either the Gaussian or admissible CDP model with a sufficiently large*

¹In the CDP model the expectation is with respect to the random modulation pattern.

number of measurements. This means that the number of samples obeys $m \geq c(\delta) \cdot n \log n$ in the Gaussian model and the number of patterns obeys $L \geq c(\delta) \cdot \log^3 n$ in the CDP model. Then

$$\|\nabla^2 f(\mathbf{x}) - \mathbb{E}[\nabla^2 f(\mathbf{x})]\| \leq \delta, \quad (15.3.7)$$

holds with probability at least $1 - 10e^{-\gamma n} - 8/n^2$ and $1 - (2L+1)/n^3$ for the Gaussian and CDP models, respectively.

We will also make use of the two results below, which are corollaries of the three lemmas above. These corollaries are also proven in Appendix H.

Corollary 15.3.5 Suppose $\|\nabla^2 f(\mathbf{x}) - \mathbb{E}[\nabla^2 f(\mathbf{x})]\| \leq \delta$. Then for all $\mathbf{h} \in \mathbb{C}^n$ obeying $\|\mathbf{h}\|_{\ell_2} = 1$, we have

$$\frac{1}{m} \sum_{r=1}^m \operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x})^2 = \frac{1}{4} \sum_{r=1}^m \begin{bmatrix} \mathbf{h} \\ \bar{\mathbf{h}} \end{bmatrix}^* \nabla^2 f(\mathbf{x}) \begin{bmatrix} \mathbf{h} \\ \bar{\mathbf{h}} \end{bmatrix} \leq \left(\frac{1}{2} \|\mathbf{h}\|_{\ell_2}^2 + \frac{3}{2} \operatorname{Re}(\mathbf{x}^* \mathbf{h})^2 - \frac{1}{2} \operatorname{Im}(\mathbf{x}^* \mathbf{h})^2 \right) + \frac{\delta}{2}.$$

In the other direction,

$$\frac{1}{m} \sum_{r=1}^m \operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x})^2 \geq \left(\frac{1}{2} \|\mathbf{h}\|_{\ell_2}^2 + \frac{3}{2} \operatorname{Re}(\mathbf{x}^* \mathbf{h})^2 - \frac{1}{2} \operatorname{Im}(\mathbf{x}^* \mathbf{h})^2 \right) - \frac{\delta}{2}.$$

Corollary 15.3.6 Suppose $\|\nabla^2 f(\mathbf{x}) - \mathbb{E}[\nabla^2 f(\mathbf{x})]\| \leq \delta$. Then for all $\mathbf{h} \in \mathbb{C}^n$ obeying $\|\mathbf{h}\|_{\ell_2} = 1$, we have

$$\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{x}|^2 |\mathbf{a}_r^* \mathbf{h}|^2 = \mathbf{h}^* \left(\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{x}|^2 \mathbf{a}_r \mathbf{a}_r^* \right) \mathbf{h} \geq (1 - \delta) \|\mathbf{h}\|_{\ell_2}^2 + |\mathbf{h}^* \mathbf{x}|^2 \geq (1 - \delta) \|\mathbf{h}\|_{\ell_2}^2,$$

and

$$\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{x}|^2 |\mathbf{a}_r^* \mathbf{h}|^2 = \mathbf{h}^* \left(\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{x}|^2 \mathbf{a}_r \mathbf{a}_r^* \right) \mathbf{h} \leq (1 + \delta) \|\mathbf{h}\|_{\ell_2}^2 + |\mathbf{h}^* \mathbf{x}|^2 \leq (2 + \delta) \|\mathbf{h}\|_{\ell_2}^2.$$

The next lemma establishes the concentration of the gradient around its mean for both Gaussian and admissible CDP models.

Lemma 15.3.7 *In the setup of Lemma 15.3.4, let $\mathbf{z} \in \mathbb{C}^n$ be a fixed vector independent of the sampling vectors obeying $\text{dist}(\mathbf{z}, \mathbf{x}) \leq \frac{1}{2}$. Then*

$$\|\nabla f(\mathbf{z}) - \mathbb{E}[\nabla f(\mathbf{z})]\|_{\ell_2} \leq \delta \cdot \text{dist}(\mathbf{z}, \mathbf{x}).$$

holds with probability at least $1 - 20e^{-\gamma m} - 4m/n^4$ in the Gaussian model and $1 - (4L + 2)/n^3$ in the CDP model.

We finish with a result concerning the concentration of the sample covariance matrix.

Lemma 15.3.8 *In the setup of Lemma 15.3.4,*

$$\left\| \mathbf{I}_n - m^{-1} \sum_{r=1}^m \mathbf{a}_r \mathbf{a}_r^* \right\| \leq \delta,$$

holds with probability at least $1 - 2e^{-\gamma m}$ for the Gaussian model and $1 - 1/n^2$ in the CDP model. On this event,

$$(1 - \delta) \|\mathbf{h}\|_{\ell_2}^2 \leq \frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^2 \leq (1 + \delta) \|\mathbf{h}\|_{\ell_2}^2 \quad \text{for all } \mathbf{h} \in \mathbb{C}^n. \quad (15.3.8)$$

15.3.4 General convergence analysis

We will assume that the function f satisfies a regularity condition on $E(\epsilon)$, which essentially states that the gradient of the function is well behaved.

Condition 15.3.9 (Regularity Condition) *We say that the function f satisfies the regularity condition or $RC(\alpha, \beta, \epsilon)$ if for all vectors $\mathbf{z} \in E(\epsilon)$ we have*

$$\text{Re}(\langle \nabla f(\mathbf{z}), \mathbf{z} - \mathbf{x} e^{i\phi(\mathbf{z})} \rangle) \geq \frac{1}{\alpha} \text{dist}^2(\mathbf{z}, \mathbf{x}) + \frac{1}{\beta} \|\nabla f(\mathbf{z})\|_{\ell_2}^2. \quad (15.3.9)$$

In the lemma below we show that as long as the regularity condition holds on $E(\epsilon)$ then Wirtinger Flow starting from an initial solution in $E(\epsilon)$ converges to a global optimizer at a geometric rate. Subsequent sections shall establish that this property holds.

Lemma 15.3.10 Assume that f obeys $RC(\alpha, \beta, \epsilon)$ for all $\mathbf{z} \in E(\epsilon)$. Furthermore, suppose $\mathbf{z}_0 \in E$, and assume $0 < \mu \leq 2/\beta$. Consider the following update

$$\mathbf{z}_{\tau+1} = \mathbf{z}_\tau - \mu \nabla f(\mathbf{z}_\tau).$$

Then for all τ we have $\mathbf{z}_\tau \in E(\epsilon)$ and

$$\text{dist}^2(\mathbf{z}_\tau, \mathbf{x}) \leq \left(1 - \frac{2\mu}{\alpha}\right)^\tau \text{dist}^2(\mathbf{z}_0, \mathbf{x}).$$

We note that for $\alpha\beta < 4$, (15.3.9) holds with the direction of the inequality reversed.² Thus, if $RC(\alpha, \beta, \epsilon)$ holds, α and β must obey $\alpha\beta \geq 4$. As a result, under the stated assumptions of Lemma 15.4.6 above, the factor $1 - 2\mu/\alpha \geq 1 - 4/(\alpha\beta)$ is non-negative.

Proof The proof follows a structure similar to related results in the convex optimization literature e.g. [186, Theorem 2.1.15]. However, unlike these classical results where the goal is often to prove convergence to a unique global optimum, the objective function f does not have a unique global optimum. Indeed, in our problem, if \mathbf{x} is solution, then $e^{i\phi}\mathbf{x}$ is also solution. Hence, proper modification is required to prove convergence results.

We prove that if $\mathbf{z} \in E(\epsilon)$ then for all $0 < \mu \leq 2/\beta$

$$\mathbf{z}_+ = \mathbf{z} - \mu \nabla f(\mathbf{z})$$

obeys

$$\text{dist}^2(\mathbf{z}_+, \mathbf{x}) \leq \left(1 - \frac{2\mu}{\alpha}\right) \text{dist}^2(\mathbf{z}, \mathbf{x}). \quad (15.3.10)$$

Therefore, if $\mathbf{z} \in E(\epsilon)$ then we also have $\mathbf{z}_+ \in E(\epsilon)$. The lemma follows by inductively applying (15.3.10). Now simple algebraic manipulations together with the regularity

²One can see this by applying Cauchy-Schwarz and calculating the determinant of the resulting quadratic form.

condition (15.3.9) give

$$\begin{aligned}
\|\mathbf{z}_+ - \mathbf{x}e^{i\phi(\mathbf{z})}\|_{\ell_2}^2 &= \|\mathbf{z} - \mathbf{x}e^{i\phi(\mathbf{z})} - \mu\nabla f(\mathbf{z})\|_{\ell_2}^2 \\
&= \|\mathbf{z} - \mathbf{x}e^{i\phi(\mathbf{z})}\|_{\ell_2}^2 - 2\mu \operatorname{Re}(\langle \nabla f(\mathbf{z}), (\mathbf{z} - \mathbf{x}e^{i\phi(\mathbf{z})}) \rangle) + \mu^2 \|\nabla f(\mathbf{z})\|_{\ell_2}^2 \\
&\leq \|\mathbf{z} - \mathbf{x}e^{i\phi(\mathbf{z})}\|_{\ell_2}^2 - 2\mu \left(\frac{1}{\alpha} \|\mathbf{z} - \mathbf{x}e^{i\phi(\mathbf{z})}\|_{\ell_2}^2 + \frac{1}{\beta} \|\nabla f(\mathbf{z})\|_{\ell_2}^2 \right) + \mu^2 \|\nabla f(\mathbf{z})\|_{\ell_2}^2 \\
&= \left(1 - \frac{2\mu}{\alpha}\right) \|\mathbf{z} - \mathbf{x}e^{i\phi(\mathbf{z})}\|_{\ell_2}^2 + \mu \left(\mu - \frac{2}{\beta}\right) \|\nabla f(\mathbf{z})\|_{\ell_2}^2 \\
&\leq \left(1 - \frac{2\mu}{\alpha}\right) \|\mathbf{z} - \mathbf{x}e^{i\phi(\mathbf{z})}\|_{\ell_2}^2,
\end{aligned}$$

where the last line follows from $\mu \leq 2/\beta$. The definition of $\phi(\mathbf{z}_+)$ gives

$$\|\mathbf{z}_+ - \mathbf{x}e^{i\phi(\mathbf{z}_+)}\|_{\ell_2}^2 \leq \|\mathbf{z}_+ - \mathbf{x}e^{i\phi(\mathbf{z})}\|_{\ell_2}^2,$$

which concludes the proof. ■

15.3.5 Proof of the regularity condition

For any $\mathbf{z} \in E(\epsilon)$, we need to show that

$$\operatorname{Re}(\langle \nabla f(\mathbf{z}), \mathbf{z} - \mathbf{x}e^{i\phi(\mathbf{z})} \rangle) \geq \frac{1}{\alpha} \operatorname{dist}^2(\mathbf{z}, \mathbf{x}) + \frac{1}{\beta} \|\nabla f(\mathbf{z})\|_{\ell_2}^2. \quad (15.3.11)$$

We prove this with $\delta = 0.01$ by establishing that our gradient satisfies the local smoothness and local curvature conditions defined below. Combining both these two properties gives (15.3.11).

Condition 15.3.11 (Local Curvature Condition) *We say that the function f satisfies the local curvature condition or LCC(α, ϵ, δ) if for all vectors $\mathbf{z} \in E(\epsilon)$,*

$$\operatorname{Re}(\langle \nabla f(\mathbf{z}), \mathbf{z} - \mathbf{x}e^{i\phi(\mathbf{z})} \rangle) \geq \left(\frac{1}{\alpha} + \frac{(1-\delta)}{4} \right) \operatorname{dist}^2(\mathbf{z}, \mathbf{x}) + \frac{1}{10m} \sum_{r=1}^m |\mathbf{a}_r^*(\mathbf{z} - e^{i\phi(\mathbf{z})}\mathbf{x})|^4. \quad (15.3.12)$$

This condition essentially states that the function curves sufficiently upwards (along most directions) near the curve of global optimizers.

Condition 15.3.12 (Local Smoothness Condition) *We say that the function f satisfies the local smoothness condition or $LSC(\beta, \epsilon, \delta)$ if for all vectors $\mathbf{z} \in E(\epsilon)$ we have*

$$\|\nabla f(\mathbf{z})\|_{\ell_2}^2 \leq \beta \left(\frac{(1-\delta)}{4} \text{dist}^2(\mathbf{z}, \mathbf{x}) + \frac{1}{10m} \sum_{r=1}^m |\mathbf{a}_r^*(\mathbf{z} - e^{i\phi(\mathbf{z})}\mathbf{x})|^4 \right). \quad (15.3.13)$$

This condition essentially states that the gradient of the function is well behaved (the function does not vary too much) near the curve of global optimizers.

15.3.6 Proof of the local curvature condition

For any $\mathbf{z} \in E(\epsilon)$, we want to prove the local curvature condition (15.3.12). Recall that

$$\nabla f(\mathbf{z}) = \frac{1}{m} \sum_{r=1}^m (|\langle \mathbf{a}_r, \mathbf{z} \rangle|^2 - y_r) (\mathbf{a}_r \mathbf{a}_r^*) \mathbf{z},$$

and define $\mathbf{h} := e^{-i\phi(\mathbf{z})}\mathbf{z} - \mathbf{x}$. To establish (15.3.12) it suffices to prove that

$$\begin{aligned} \frac{1}{m} \sum_{r=1}^m (2 \operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x})^2 + 3 \operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x}) |\mathbf{a}_r^* \mathbf{h}|^2 + |\mathbf{a}_r^* \mathbf{h}|^4) - \left(\frac{1}{10m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^4 \right) \\ \geq \left(\frac{1}{\alpha} + \frac{(1-\delta)}{4} \right) \|\mathbf{h}\|_{\ell_2}^2, \end{aligned} \quad (15.3.14)$$

holds for all \mathbf{h} satisfying $\operatorname{Im}(\mathbf{h}^* \mathbf{x}) = 0$, $\|\mathbf{h}\|_2 \leq \epsilon$. Equivalently, we only need to prove that for all \mathbf{h} satisfying $\operatorname{Im}(\mathbf{h}^* \mathbf{x}) = 0$, $\|\mathbf{h}\|_2 = 1$ and for all s with $0 \leq s \leq \epsilon$,

$$\frac{1}{m} \sum_{r=1}^m \left(2 \operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x})^2 + 3s \operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x}) |\mathbf{a}_r^* \mathbf{h}|^2 + \frac{9}{10} s^2 |\mathbf{a}_r^* \mathbf{h}|^4 \right) \geq \frac{1}{\alpha} + \frac{(1-\delta)}{4}. \quad (15.3.15)$$

By Corollary 15.3.5, with high probability,

$$\frac{1}{m} \sum_{r=1}^m \operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x})^2 \leq \frac{1+\delta}{2} + \frac{3}{2} \operatorname{Re}(\mathbf{x}^* \mathbf{h})^2,$$

holds for all \mathbf{h} obeying $\|\mathbf{h}\|_{\ell_2} = 1$. Therefore, to establish the local curvature condition (15.3.12) it suffices to show that

$$\frac{1}{m} \sum_{r=1}^m \left(\frac{5}{2} \operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x})^2 + 3s \operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x}) |\mathbf{a}_r^* \mathbf{h}|^2 + \frac{9}{10} s^2 |\mathbf{a}_r^* \mathbf{h}|^4 \right) \geq \left(\frac{1}{\alpha} + \frac{1}{2} \right) + \frac{3}{4} \operatorname{Re}(\mathbf{x}^* \mathbf{h})^2. \quad (15.3.16)$$

We will establish (15.3.16) for different measurement models and different values of ϵ . Below, it shall be convenient to use the shorthand

$$\begin{aligned} Y_r(\mathbf{h}, s) &:= \frac{5}{2} \operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x})^2 + 3s \operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x}) |\mathbf{a}_r^* \mathbf{h}|^2 + \frac{9}{10} s^2 |\mathbf{a}_r^* \mathbf{h}|^4, \\ \langle Y_r(\mathbf{h}, s) \rangle &:= \frac{1}{m} \sum_{r=1}^m Y_r(\mathbf{h}, s). \end{aligned}$$

15.3.6.1 Proof of (15.3.16) with $\epsilon = 1/8\sqrt{n}$ in the Gaussian and CDP models

Set $\epsilon = 1/8\sqrt{n}$. We show that with high probability, (15.3.16) holds for all \mathbf{h} satisfying $\operatorname{Im}(\mathbf{h}^* \mathbf{x}) = 0$, $\|\mathbf{h}\|_2 = 1$, $0 \leq s \leq \epsilon$, $\delta \leq 0.01$, and $\alpha \geq 30$. First, note that by Cauchy-Schwarz inequality,

$$\begin{aligned} \langle Y_r(\mathbf{h}, s) \rangle &\geq \frac{5}{2m} \sum_{r=1}^m \operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x})^2 - \frac{3s}{m} \sqrt{\sum_{r=1}^m \operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x})^2} \sqrt{\sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^4} + \frac{9}{10} \frac{s^2}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^4 \\ &= \left(\sqrt{\frac{5}{2m} \sum_{r=1}^m \operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x})^2} - s \sqrt{\frac{9}{10m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^4} \right)^2 \\ &\geq \frac{5}{4m} \sum_{r=1}^m \operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x})^2 - \frac{9s^2}{10m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^4. \end{aligned} \quad (15.3.17)$$

The last inequality follows from $(a - b)^2 \geq \frac{a^2}{2} - b^2$. By Corollary 15.3.5,

$$\frac{1}{m} \sum_{r=1}^m \operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x})^2 \geq \frac{1-\delta}{2} + \frac{3}{2} \operatorname{Re}(\mathbf{x}^* \mathbf{h})^2 \quad (15.3.18)$$

holds with high probability for all \mathbf{h} obeying $\|\mathbf{h}\|_{\ell_2} = 1$. Furthermore, by applying Lemma 15.3.8,

$$\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^4 \leq (\max_r \|\mathbf{a}_r\|_{\ell_2}^2) \left(\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^2 \right) \leq 6(1+\delta)n \quad (15.3.19)$$

holds with high probability. Plugging (15.3.18) and (15.3.19) in (15.3.17) yields

$$\langle Y_r(\mathbf{h}, s) \rangle \geq \frac{15}{8} \operatorname{Re}(\mathbf{x}^* \mathbf{h})^2 + \frac{5}{8}(1-\delta) - \frac{27}{5}s^2(1+\delta)n.$$

(15.3.16) follows by using $\alpha \geq 30$, $\epsilon = \frac{1}{8\sqrt{n}}$ and $\delta = 0.01$.

15.3.6.2 Proof of (15.3.16) with $\epsilon = 1/8$ in the Gaussian model

Set $\epsilon = 1/8$. We show that with high probability, (15.3.16) holds for all \mathbf{h} satisfying $\operatorname{Im}(\mathbf{h}^* \mathbf{x}) = 0$, $\|\mathbf{h}\|_2 = 1$, $0 \leq s \leq \epsilon$, $\delta \leq 2$, and $\alpha \geq 8$. To this end, we first state a result about the tail of a sum of i.i.d. random variables. Below, Φ is the cumulative distribution function of a standard normal variable.

Lemma 15.3.13 ([36]) *Suppose X_1, X_2, \dots, X_m are i.i.d. real-valued random variables obeying $X_r \leq b$ for some nonrandom $b > 0$, $\mathbb{E} X_r = 0$, and $\mathbb{E} X_r^2 = v^2$. Setting $\sigma^2 = m \max(b^2, v^2)$,*

$$\mathbb{P}(X_1 + \dots + X_m \geq y) \leq \min \left(\exp \left(-\frac{y^2}{2\sigma^2} \right), c_0 (1 - \Phi(y/\sigma)) \right)$$

where one can take $c_0 = 25$.

To establish (15.3.16) we first prove it for a fixed \mathbf{h} , and then use a covering argument. Observe that

$$Y_r := Y_r(\mathbf{h}, s) = \left(\sqrt{\frac{5}{2}} \operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x}) + \sqrt{\frac{9}{10}} s |\mathbf{a}_r^* \mathbf{h}|^2 \right)^2.$$

By Lemma 15.3.3,

$$\mathbb{E}[\operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x})^2] = \frac{1}{2} + \frac{3}{2} (\operatorname{Re}(\mathbf{x}^* \mathbf{h}))^2 \text{ and } \mathbb{E}[\operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x}) |\mathbf{a}_r^* \mathbf{h}|^2] = 2 \operatorname{Re}(\mathbf{u}^* \mathbf{v}),$$

compare (15.3.4) and (15.3.5). Therefore, using $s \leq \frac{1}{8}$,

$$\mu_r = \mathbb{E} Y_r = \frac{5}{4} (1 + 3 \operatorname{Re}(\mathbf{x}^* \mathbf{h})^2) + 6s \operatorname{Re}(\mathbf{x}^* \mathbf{h}) + \frac{27}{10} s^2 < 6.$$

Now define $X_r = \mu_r - Y_r$. First, since $Y_r \geq 0$, $X_r \leq \mu_r < 6$. Second, we bound $\mathbb{E} X_r^2$ using Lemma 15.3.3 and Holder's inequality with $s \leq 1/8$:

$$\begin{aligned} \mathbb{E} X_r^2 &\leq \mathbb{E} Y_r^2 = \frac{25}{4} \mathbb{E}[\operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x})^4] + \frac{81}{100} s^4 \mathbb{E}[|\mathbf{a}_r^* \mathbf{h}|^8] + \frac{27}{2} s^2 \mathbb{E}[\operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x})^2 |\mathbf{a}_r^* \mathbf{h}|^4] \\ &\quad + 15s \mathbb{E}[\operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x})^3 |\mathbf{a}_r^* \mathbf{h}|^2] + \frac{27}{5} s^3 \mathbb{E}[\operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x}) |\mathbf{a}_r^* \mathbf{h}|^6] \\ &\leq \frac{25}{4} \sqrt{\mathbb{E}[|\mathbf{a}_r^* \mathbf{h}|^8] \mathbb{E}[|\mathbf{a}_r^* \mathbf{x}|^8]} + \frac{81}{100} s^4 \mathbb{E}[|\mathbf{a}_r^* \mathbf{h}|^8] + \frac{27}{2} s^2 \sqrt{\mathbb{E}[|\mathbf{a}_r^* \mathbf{h}|^{12}] \mathbb{E}[|\mathbf{a}_r^* \mathbf{x}|^4]} \\ &\quad + 15s \sqrt{\mathbb{E}[|\mathbf{a}_r^* \mathbf{h}|^{10}] \mathbb{E}[|\mathbf{a}_r^* \mathbf{x}|^6]} + \frac{27}{5} s^3 \sqrt{\mathbb{E}[|\mathbf{a}_r^* \mathbf{h}|^{14}] \mathbb{E}[|\mathbf{a}_r^* \mathbf{x}|^2]} \\ &< 20s^4 + 543s^3 + 513s^2 + 403s + 150 \\ &< 210. \end{aligned}$$

We have all the elements to apply Lemma 15.4.9 with $\sigma^2 = m \max(9^2, 210) = 210m$ and $y = m/4$:

$$\mathbb{P}\left(m\mu - \sum_{r=1}^m Y_r \geq \frac{m}{4}\right) \leq e^{-2\gamma m}$$

with $\gamma = 1/840$. Therefore, with probability at least $1 - e^{-2\gamma m}$, we have

$$\begin{aligned} \langle Y_r(\mathbf{h}, s) \rangle &\geq \frac{5}{4}(1 + 3 \operatorname{Re}(\mathbf{x}^* \mathbf{h})^2) + 6s \operatorname{Re}(\mathbf{x}^* \mathbf{h}) + 2.7s^2 - \frac{1}{4} \\ &\geq \frac{3}{4} + \frac{3}{4} \operatorname{Re}(\mathbf{x}^* \mathbf{h})^2 + 3(\operatorname{Re}(\mathbf{x}^* \mathbf{h}) + s)^2 + \left(\frac{1}{4} - \frac{3}{10}s^2 \right) \\ &\geq \frac{3}{4} + \frac{3}{4} \operatorname{Re}(\mathbf{x}^* \mathbf{h})^2. \end{aligned} \quad (15.3.20)$$

provided that $s \leq \sqrt{5/6}$. The inequality above holds for a fixed vector \mathbf{h} . To prove (15.3.16) for all $\mathbf{h} \in \mathbb{C}^n$ with $\|\mathbf{h}\|_{\ell_2} = 1$, define

$$p_r(\mathbf{h}) := \sqrt{\frac{5}{2}} \operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x}) + \sqrt{\frac{9}{10}} s |\mathbf{a}_r^* \mathbf{h}|^2.$$

Using the fact that $\max_r |\mathbf{a}_r| \leq \sqrt{6n}$ and $s \leq 1/8$, we have $|p_r(\mathbf{h})| \leq 2 |\mathbf{a}_r^* \mathbf{h}| |\mathbf{a}_r^* \mathbf{x}| + s |\mathbf{a}_r^* \mathbf{h}|^2 \leq 13n$. Moreover, for any $\mathbf{u}, \mathbf{v} \in \mathbb{C}^n$ obeying $\|\mathbf{u}\|_{\ell_2} = \|\mathbf{v}\|_{\ell_2} = 1$,

$$|p_r(\mathbf{u}) - p_r(\mathbf{v})| \leq \left| \sqrt{\frac{5}{2}} \operatorname{Re}((\mathbf{u} - \mathbf{v})^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x}) \right| + \sqrt{\frac{9}{10}} s |\mathbf{a}_r^*(\mathbf{u} + \mathbf{v})| |\mathbf{a}_r^*(\mathbf{u} - \mathbf{v})| \leq \frac{27}{2} n \|\mathbf{u} - \mathbf{v}\|_{\ell_2}.$$

Introduce

$$q(\mathbf{h}) := \frac{1}{m} \sum_{r=1}^m p_r(\mathbf{h})^2 - \frac{3}{4} \operatorname{Re}(\mathbf{x}^* \mathbf{h})^2 = \langle Y_r(\mathbf{h}, s) \rangle - \frac{3}{4} \operatorname{Re}(\mathbf{x}^* \mathbf{h})^2.$$

For any $\mathbf{u}, \mathbf{v} \in \mathbb{C}^n$ obeying $\|\mathbf{u}\|_{\ell_2} = \|\mathbf{v}\|_{\ell_2} = 1$,

$$\begin{aligned} |q(\mathbf{u}) - q(\mathbf{v})| &= \left| \frac{1}{m} \sum_{r=1}^m (p_r(\mathbf{u}) - p_r(\mathbf{v}))(p_r(\mathbf{u}) + p_r(\mathbf{v})) - \frac{3}{4} \operatorname{Re}(\mathbf{x}^*(\mathbf{u} - \mathbf{v})) \operatorname{Re}(\mathbf{x}^*(\mathbf{u} + \mathbf{v})) \right| \\ &\leq \frac{27n}{2} \times 2 \times 13n \|\mathbf{u} - \mathbf{v}\|_{\ell_2} + \frac{3}{2} \|\mathbf{u} - \mathbf{v}\|_{\ell_2} \\ &= \left(351n^2 + \frac{3}{2} \right) \|\mathbf{u} - \mathbf{v}\|_{\ell_2}. \end{aligned} \quad (15.3.21)$$

Therefore, for any $\mathbf{u}, \mathbf{v} \in \mathbb{C}^n$ obeying $\|\mathbf{u}\|_{\ell_2} = \|\mathbf{v}\|_{\ell_2} = 1$ and $\|\mathbf{u} - \mathbf{v}\|_{\ell_2} \leq \eta := \frac{1}{6000n^2}$, we have

$$q(\mathbf{v}) \geq q(\mathbf{u}) - \frac{1}{16}. \quad (15.3.22)$$

Let \mathcal{N}_η be an η -net for the unit sphere of \mathbb{C}^n with cardinality obeying $|\mathcal{N}_\eta| \leq (1 + \frac{2}{\eta})^{2n}$. Applying (15.3.20) together with the union bound we conclude that for all $\mathbf{u} \in \mathcal{N}_\eta$

$$\begin{aligned} \mathbb{P}\left(q(\mathbf{u}) \geq \frac{3}{4}\right) &\geq 1 - |\mathcal{N}_\eta|e^{-2\gamma m} \\ &\geq 1 - (1 + 12000n^2)^n e^{-2\gamma m} \\ &\geq 1 - e^{-\gamma m}. \end{aligned} \quad (15.3.23)$$

The last line follows by choosing m such that $m \geq c \cdot n \log n$, where c is a sufficiently large constant. Now for any \mathbf{h} on the unit sphere of \mathbb{C}^n , there exists a vector $\mathbf{u} \in \mathcal{N}_\eta$ such that $\|\mathbf{h} - \mathbf{u}\|_{\ell_2} \leq \eta$. By combining (15.3.22) and (15.3.23),

$$q(\mathbf{h}) \geq \frac{3}{4} - \frac{1}{16} > \frac{5}{8} \quad \Rightarrow \quad \langle Y_r(\mathbf{h}, s) \rangle \geq \left(\frac{1}{8} + \frac{1}{2}\right) + \frac{3}{4} \operatorname{Re}(\mathbf{x}^* \mathbf{h})^2,$$

holds with probability at least $1 - e^{-\gamma m}$. This concludes the proof of (15.3.16) with $\alpha \geq 8$.

15.3.7 Proof of the local smoothness condition

For any $\mathbf{z} \in E(\epsilon)$, we want to prove (15.3.13), which is equivalent to proving that for all $\mathbf{u} \in \mathbb{C}^n$ obeying $\|\mathbf{u}\|_{\ell_2} = 1$, we have

$$|(\nabla f(\mathbf{z}))^* \mathbf{u}|^2 \leq \beta \left(\frac{(1-\delta)}{4} \operatorname{dist}^2(\mathbf{z}, \mathbf{x}) + \frac{1}{10m} \sum_{r=1}^m |\mathbf{a}_r^*(\mathbf{z} - e^{i\phi(\mathbf{z})} \mathbf{x})|^4 \right).$$

Recall that

$$\nabla f(\mathbf{z}) = \frac{1}{m} \sum_{r=1}^m (|\langle \mathbf{a}_r, \mathbf{z} \rangle|^2 - y_r) (\mathbf{a}_r \mathbf{a}_r^*) \mathbf{z}$$

and define

$$\begin{aligned} g(\mathbf{h}, \mathbf{w}, s) = & \frac{1}{m} \sum_{r=1}^m \left(2 \operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x}) \operatorname{Re}(\mathbf{w}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x}) + s |\mathbf{a}_r^* \mathbf{h}|^2 \operatorname{Re}(\mathbf{w}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x}) \right. \\ & \left. + 2s \operatorname{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x}) \operatorname{Re}(\mathbf{w}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{h}) + s^2 |\mathbf{a}_r^* \mathbf{h}|^2 \operatorname{Re}(\mathbf{w}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{h}) \right). \end{aligned}$$

Define $\mathbf{h} := e^{-i\phi_z} \mathbf{z} - \mathbf{x}$ and $\mathbf{w} := e^{-i\phi_z} \mathbf{u}$, to establish (15.3.13) it suffices to prove that

$$|g(\mathbf{h}, \mathbf{w}, 1)|^2 \leq \beta \left(\frac{1-\delta}{4} \|\mathbf{h}\|_{\ell_2}^2 + \frac{1}{10m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^4 \right). \quad (15.3.24)$$

holds for all \mathbf{h} and \mathbf{w} satisfying $\operatorname{Im}(\mathbf{h}^* \mathbf{x}) = 0$, $\|\mathbf{h}\|_{\ell_2} \leq \epsilon$ and $\|\mathbf{w}\|_{\ell_2} = 1$. Equivalently, we only need to prove for all \mathbf{h} and \mathbf{w} satisfying $\operatorname{Im}(\mathbf{h}^* \mathbf{x}) = 0$, $\|\mathbf{h}\|_{\ell_2} = \|\mathbf{w}\|_{\ell_2} = 1$ and $\forall s : 0 \leq s \leq \epsilon$,

$$|g(\mathbf{h}, \mathbf{w}, s)|^2 \leq \beta \left(\frac{1-\delta}{4} + \frac{s^2}{10m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^4 \right). \quad (15.3.25)$$

Note that since $(a+b+c)^2 \leq 3(a^2 + b^2 + c^2)$

$$\begin{aligned} |g(\mathbf{h}, \mathbf{w}, s)|^2 & \leq \left| \frac{1}{m} \sum_{r=1}^m \left(2 |\mathbf{h}^* \mathbf{a}_r| |\mathbf{w}^* \mathbf{a}_r| |\mathbf{a}_r^* \mathbf{x}|^2 + 3s |\mathbf{h}^* \mathbf{a}_r|^2 |\mathbf{a}_r^* \mathbf{x}| |\mathbf{w}^* \mathbf{a}_r| + s^2 |\mathbf{a}_r^* \mathbf{h}|^3 |\mathbf{w}^* \mathbf{a}_r| \right) \right|^2 \\ & \leq 3 \left| \frac{2}{m} \sum_{r=1}^m |\mathbf{h}^* \mathbf{a}_r| |\mathbf{w}^* \mathbf{a}_r| |\mathbf{a}_r^* \mathbf{x}|^2 \right|^2 + 3 \left| \frac{3s}{m} \sum_{r=1}^m |\mathbf{h}^* \mathbf{a}_r|^2 |\mathbf{a}_r^* \mathbf{x}| |\mathbf{w}^* \mathbf{a}_r| \right|^2 + 3 \left| \frac{s^2}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^3 |\mathbf{w}^* \mathbf{a}_r| \right|^2 \\ & := 3(I_1 + I_2 + I_3). \end{aligned} \quad (15.3.26)$$

We now bound each of the terms on the right-hand side. For the first term we use Cauchy-Schwarz and Corollary 15.3.6, which give

$$I_1 \leq \left(\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{x}|^2 |\mathbf{a}_r^* \mathbf{w}|^2 \right) \left(\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{x}|^2 |\mathbf{a}_r^* \mathbf{h}|^2 \right) \leq (2+\delta)^2. \quad (15.3.27)$$

Similarly, for the second term, we have

$$I_2 \leq \left(\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^4 \right) \left(\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{w}|^2 |\mathbf{a}_r^* \mathbf{x}|^2 \right) \leq \frac{2+\delta}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^4. \quad (15.3.28)$$

Finally, for the third term we use the Cauchy-Schwarz inequality together with Lemma 15.3.8 (inequality) (15.3.8) to derive

$$\begin{aligned} I_3 &\leq \left(\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^3 \max_r \|\mathbf{a}_r\|_{\ell_2} \right)^2 \leq 6n \left(\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^3 \right)^2 \\ &\leq 6n \left(\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^4 \right) \left(\sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^2 \right) \\ &\leq \frac{6n(1+\delta)}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^4. \end{aligned} \quad (15.3.29)$$

We now plug these inequalities into (15.3.26) and get

$$\begin{aligned} |g(\mathbf{h}, \mathbf{w}, s)|^2 &\leq 12(2+\delta)^2 + \frac{27s^2(2+\delta)}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^4 + \frac{18s^4n(1+\delta)}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^4 \\ &\leq \beta \left(\frac{1-\delta}{4} + \frac{s^2}{10m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^4 \right), \end{aligned} \quad (15.3.30)$$

which completes the proof of (15.3.25) and, in turn, establishes the local smoothness condition in (15.3.13). However, the last line of (15.3.30) holds as long as

$$\beta \geq \max \left(48 \frac{(2+\delta)^2}{1-\delta}, 270(2+\delta) + 180\epsilon^2 n(1+\delta) \right). \quad (15.3.31)$$

In our theorems we use two different values of $\epsilon = \frac{1}{8\sqrt{n}}$ and $\epsilon = \frac{1}{8}$. Using $\delta \leq 0.01$ in (15.3.31) we conclude that the local smoothness condition (15.3.30) holds as long as

$$\begin{aligned} \beta &\geq 550 & \text{for } \epsilon &= 1/(8\sqrt{n}), \\ \beta &\geq 3n + 550 & \text{for } \epsilon &= 1/8. \end{aligned}$$

15.3.8 Wirtinger flow initialization

In this section, we prove that the WF initialization obeys (12.2.1) from Theorem 12.2.3. Recall that

$$\mathbf{Y} := \frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{x}|^2 \mathbf{a}_r \mathbf{a}_r^*.$$

and that Lemma 15.3.4 gives

$$\|\mathbf{Y} - (\mathbf{x}\mathbf{x}^* + \|\mathbf{x}\|_{\ell_2}^2 \mathbf{I})\| \leq \delta := 0.001.$$

Let $\tilde{\mathbf{z}}_0$ be the eigenvector corresponding to the top eigenvalue λ_0 of \mathbf{Y} . It is easy to see that

$$|\lambda_0 - (|\tilde{\mathbf{z}}_0^* \mathbf{x}|^2 + 1)| = |\tilde{\mathbf{z}}_0^* \mathbf{Y} \tilde{\mathbf{z}}_0 - \tilde{\mathbf{z}}_0^* (\mathbf{x}\mathbf{x}^* + \mathbf{I}) \tilde{\mathbf{z}}_0| = |\tilde{\mathbf{z}}_0^* (\mathbf{Y} - (\mathbf{x}\mathbf{x}^* + \mathbf{I})) \tilde{\mathbf{z}}_0| \leq \|\mathbf{Y} - (\mathbf{x}\mathbf{x}^* + \mathbf{I})\| \leq \delta.$$

Therefore,

$$|\tilde{\mathbf{z}}_0^* \mathbf{x}|^2 \geq \lambda_0 - 1 - \delta.$$

Also, since λ_0 is the top eigenvalue of \mathbf{Y} , and $\|\mathbf{x}\|_{\ell_2} = 1$, we have

$$\lambda_0 \geq \mathbf{x}^* \mathbf{Y} \mathbf{x} = \mathbf{x}^* (\mathbf{Y} - (\mathbf{I} + \mathbf{x}\mathbf{x}^*)) \mathbf{x} + 2 \geq 2 - \delta.$$

Combining the above two inequalities together, we have

$$|\tilde{\mathbf{z}}_0^* \mathbf{x}|^2 \geq 1 - 2\delta \quad \Rightarrow \quad \text{dist}^2(\tilde{\mathbf{z}}_0, \mathbf{x}) \leq 2 - 2\sqrt{1 - 2\delta} = \frac{1}{256} \quad \Rightarrow \quad \text{dist}(\tilde{\mathbf{z}}_0, \mathbf{x}) \leq \frac{1}{16}.$$

Recall that $\mathbf{z}_0 = \left(\sqrt{\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{x}|^2} \right) \tilde{\mathbf{z}}_0$. By Lemma 15.3.8, equation (15.3.8), with high probability we have

$$|\|\mathbf{z}_0\|_{\ell_2} - 1| = \left| \frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{x}|^2 \right| \leq \delta < \frac{1}{16}.$$

Therefore, we have

$$\text{dist}(\mathbf{z}_0, \mathbf{x}) \leq \|\mathbf{z}_0 - \tilde{\mathbf{z}}_0\|_{\ell_2} + \text{dist}(\tilde{\mathbf{z}}_0, \mathbf{x}) = |\|\mathbf{z}_0\|_{\ell_2} - 1| + \text{dist}(\tilde{\mathbf{z}}_0, \mathbf{x}) \leq \frac{1}{8}.$$

15.3.9 Initialization via resampled Wirtinger Flow

In this section, we prove that the output of Algorithm 8 obeys (12.2.2) from Theorem 12.2.4. Introduce

$$F(\mathbf{z}) = \frac{1}{2} \mathbf{z}^* (\mathbf{I} - \mathbf{x} \mathbf{x}^*) \mathbf{z} + (\|\mathbf{z}\|_{\ell_2}^2 - 1)^2.$$

By Lemma 15.3.2, if $\mathbf{z} \in \mathbb{C}^n$ is a vector independent from the measurements, then

$$\mathbb{E} \nabla f(\mathbf{z}; b) = \nabla F(\mathbf{z}).$$

We prove that F obeys a regularization condition in $E(1/8)$, namely,

$$\operatorname{Re}(\langle \nabla F(\mathbf{z}), \mathbf{z} - \mathbf{x} e^{i\phi(\mathbf{z})} \rangle) \geq \frac{1}{\alpha'} \operatorname{dist}^2(\mathbf{z}, \mathbf{x}) + \frac{1}{\beta'} \|\nabla F(\mathbf{z})\|_{\ell_2}^2. \quad (15.3.32)$$

Lemma 15.3.7 implies that for a fixed vector \mathbf{z} ,

$$\begin{aligned} \operatorname{Re}(\langle \nabla f(\mathbf{z}; b), \mathbf{z} - \mathbf{x} e^{i\phi(\mathbf{z})} \rangle) &= \operatorname{Re}(\langle \nabla F(\mathbf{z}), \mathbf{z} - \mathbf{x} e^{i\phi(\mathbf{z})} \rangle) + \operatorname{Re}(\langle \nabla f(\mathbf{z}; b) - \nabla F(\mathbf{z}), \mathbf{z} - \mathbf{x} e^{i\phi(\mathbf{z})} \rangle) \\ &\geq \operatorname{Re}(\langle \nabla F(\mathbf{z}), \mathbf{z} - \mathbf{x} e^{i\phi(\mathbf{z})} \rangle) - \|\nabla f(\mathbf{z}; b) - \nabla F(\mathbf{z})\|_{\ell_2} \operatorname{dist}(\mathbf{z}, \mathbf{x}) \\ &\geq \operatorname{Re}(\langle \nabla F(\mathbf{z}), \mathbf{z} - \mathbf{x} e^{i\phi(\mathbf{z})} \rangle) - \delta \operatorname{dist}(\mathbf{z}, \mathbf{x})^2 \\ &\geq \left(\frac{1}{\alpha'} - \delta \right) \operatorname{dist}(\mathbf{z}, \mathbf{x})^2 + \frac{1}{\beta'} \|\nabla F(\mathbf{z})\|_{\ell_2}^2, \end{aligned} \quad (15.3.33)$$

holds with high probability. The last inequality follows from (15.3.32). Applying Lemma 15.3.7, we also have

$$\|\nabla F(\mathbf{z})\|_{\ell_2}^2 \geq \frac{1}{2} \|\nabla f(\mathbf{z}; b)\|_{\ell_2}^2 - \|\nabla f(\mathbf{z}; b) - \nabla F(\mathbf{z})\|_{\ell_2}^2 \geq \frac{1}{2} \|\nabla f(\mathbf{z}; b)\|_{\ell_2}^2 - \delta^2 \operatorname{dist}(\mathbf{z}, \mathbf{x})^2.$$

Plugging the latter into (15.3.33) yields

$$\begin{aligned} \operatorname{Re}(\langle \nabla f(\mathbf{z}; b), \mathbf{z} - \mathbf{x} e^{i\phi(\mathbf{z})} \rangle) &\geq \left(\frac{1}{\alpha'} - \frac{\delta^2}{\beta'} - \delta \right) \operatorname{dist}^2(\mathbf{z}, \mathbf{x}) + \frac{1}{2\beta'} \|\nabla f(\mathbf{z}; b)\|_{\ell_2}^2 \\ &:= \frac{1}{\tilde{\alpha}} \operatorname{dist}^2(\mathbf{z}, \mathbf{x}) + \frac{1}{\tilde{\beta}} \|\nabla f(\mathbf{z}; b)\|_{\ell_2}^2. \end{aligned}$$

Therefore, using the general convergence analysis of gradient descent discussed in Section 15.3.4 we conclude that for all $\tilde{\mu} \leq \tilde{\mu}_0 := 2/\tilde{\beta}$,

$$\text{dist}^2(\mathbf{u}_{b+1}, \mathbf{x}) \leq \left(1 - \frac{2\tilde{\mu}}{\tilde{\alpha}}\right) \text{dist}^2(\mathbf{u}_b, \mathbf{x}).$$

Finally,

$$B \geq -\frac{\log n}{\log\left(1 - \frac{2\tilde{\mu}}{\tilde{\alpha}}\right)} \implies \text{dist}(\mathbf{u}_B, \mathbf{x}) \leq \left(1 - \frac{2\tilde{\mu}}{\tilde{\alpha}}\right)^{\frac{B}{2}} \text{dist}(\mathbf{u}_0, \mathbf{x}) \leq \left(1 - \frac{2\tilde{\mu}}{\tilde{\alpha}}\right)^{\frac{B}{2}} \frac{1}{8} \leq \frac{1}{8\sqrt{n}}.$$

It only remains to establish (15.3.32). First, without loss of generality, we can assume that $\phi(\mathbf{z}) = 0$, which implies $\text{Re}(\mathbf{z}^* \mathbf{x}) = |\mathbf{z}^* \mathbf{x}|$ and use $\|\mathbf{z} - \mathbf{x}\|_{\ell_2}$ in lieu of $\text{dist}(\mathbf{z}, \mathbf{x})$. Set $\mathbf{h} := \mathbf{z} - \mathbf{x}$ so that $\text{Im}(\mathbf{x}^* \mathbf{h}) = 0$. This implies

$$\begin{aligned} \nabla F(\mathbf{z}) &= (\mathbf{I} - \mathbf{x}\mathbf{x}^*)\mathbf{z} + 2(\|\mathbf{z}\|_{\ell_2}^2 - 1)\mathbf{z} \\ &= (\mathbf{I} - \mathbf{x}\mathbf{x}^*)(\mathbf{x} + \mathbf{h}) + 2(\|\mathbf{x} + \mathbf{h}\|_{\ell_2}^2 - 1)(\mathbf{x} + \mathbf{h}) \\ &= (\mathbf{I} - \mathbf{x}\mathbf{x}^*)\mathbf{h} + 2(2\text{Re}(\mathbf{x}^* \mathbf{h}) + \|\mathbf{h}\|_{\ell_2}^2)(\mathbf{x} + \mathbf{h}) \\ &= (1 + 4(\mathbf{x}^* \mathbf{h}) + 2\|\mathbf{h}\|_{\ell_2}^2)\mathbf{h} + (3(\mathbf{x}^* \mathbf{h}) + 2\|\mathbf{h}\|_{\ell_2}^2)\mathbf{x}. \end{aligned}$$

Therefore,

$$\|\nabla F(\mathbf{z})\|_{\ell_2} \leq 4\|\mathbf{h}\|_{\ell_2} + 6\|\mathbf{h}\|_{\ell_2}^2 + 2\|\mathbf{h}\|_{\ell_2}^3 \leq 5\|\mathbf{h}\|_{\ell_2}, \quad (15.3.34)$$

where the last inequality is due to $\|\mathbf{h}\|_{\ell_2} \leq \epsilon \leq 1/8$. Furthermore,

$$\begin{aligned} \text{Re}(\langle \nabla F(\mathbf{z}), \mathbf{z} - \mathbf{x} \rangle) &= \text{Re}(\langle (1 + 4(\mathbf{x}^* \mathbf{h}) + 2\|\mathbf{h}\|_{\ell_2}^2)\mathbf{h} + (3(\mathbf{x}^* \mathbf{h}) + 2\|\mathbf{h}\|_{\ell_2}^2)\mathbf{x}, \mathbf{h} \rangle) \\ &= \|\mathbf{h}\|_{\ell_2}^2 + 2\|\mathbf{h}\|_{\ell_2}^4 + 6\|\mathbf{h}\|_{\ell_2}^2(\mathbf{x}^* \mathbf{h}) + 3|\mathbf{x}^* \mathbf{h}|^2 \geq \frac{1}{4}\|\mathbf{h}\|_{\ell_2}^2, \quad (15.3.35) \end{aligned}$$

where the last inequality also holds because $\|\mathbf{h}\|_{\ell_2} \leq \epsilon \leq 1/8$. Finally, (15.3.34) and (15.3.35) imply

$$\text{Re}(\langle \nabla F(\mathbf{z}), \mathbf{z} - \mathbf{x} \rangle) \geq \frac{1}{4}\|\mathbf{h}\|_{\ell_2}^2 \geq \frac{1}{8}\|\mathbf{h}\|_{\ell_2}^2 + \frac{1}{200}\|\nabla F(\mathbf{z})\|_{\ell_2}^2 := \frac{1}{\alpha'}\|\mathbf{h}\|_{\ell_2}^2 + \frac{1}{\beta'}\|\nabla F(\mathbf{z})\|_{\ell_2}^2,$$

where $\alpha' = 8$ and $\beta' = 200$.

15.4 Proofs of stability of Wirtinger flow

In this section we prove the results on stability of Wirtinger flow. In the first subsection we prove Theorem 12.3.1 and then prove Theorem 12.3.2 in the next two subsections.

15.4.1 Proof of stability of the global optimum of the WF objective (Proof of Theorem 12.3.1)

To establish Theorem 12.3.1 we first state a lemma. The proof of this lemma is based on tools for bounding non-negative empirical processes. Due to time constraints we defer the proof to a future publication.

Lemma 15.4.1 *Let \mathbf{x} be an arbitrary vector in \mathbb{C}^n and let $\{\mathbf{a}_r\}_{r=1}^m$ be the Gaussian sensing vectors with $m \geq c_0 \cdot n$, where c_0 is a sufficiently large numerical constant. Then for all $\mathbf{x} \in \mathbb{C}^n$, the solution to (12.3.1) obeys*

$$\frac{1}{m} \sum_{r=1}^m (|\mathbf{a}_r^* \mathbf{x}_2|^2 - |\mathbf{a}_r^* \mathbf{x}_1|^2)^2 \geq 0.9124 \|\mathbf{x}_2 \mathbf{x}_2^* - \mathbf{x}_1 \mathbf{x}_1^*\|^2.$$

Turning our attention to the proof of the theorem by the optimality of $\hat{\mathbf{x}}$ for (12.3.1) we have

$$f(\hat{\mathbf{x}}) \leq f(\mathbf{x}) = \frac{\|\mathbf{w}\|_{\ell_2}^2}{m}.$$

Therefore,

$$\frac{1}{m} \sum_{r=1}^m (|\mathbf{a}_r^* \hat{\mathbf{x}}|^2 - |\mathbf{a}_r^* \mathbf{x}|^2 - w_r)^2 \leq \frac{\|\mathbf{w}\|_{\ell_2}^2}{m}.$$

Rearranging the terms and using Cauchy-Schwarz we arrive at

$$\frac{1}{m} \sum_{r=1}^m (|\mathbf{a}_r^* \hat{\mathbf{x}}|^2 - |\mathbf{a}_r^* \mathbf{x}|^2)^2 \leq \frac{2}{m} \sum_{r=1}^m w_r (|\mathbf{a}_r^* \hat{\mathbf{x}}|^2 - |\mathbf{a}_r^* \mathbf{x}|^2) \leq \frac{2 \|\mathbf{w}\|_{\ell_2}}{\sqrt{m}} \sqrt{\frac{1}{m} \sum_{r=1}^m (|\mathbf{a}_r^* \hat{\mathbf{x}}|^2 - |\mathbf{a}_r^* \mathbf{x}|^2)^2}.$$

Thus,

$$\frac{1}{m} \sum_{r=1}^m (|\mathbf{a}_r^* \hat{\mathbf{x}}|^2 - |\mathbf{a}_r^* \mathbf{x}|^2)^2 \leq 4 \frac{\|\mathbf{w}\|_{\ell_2}^2}{m}.$$

The result follows from Lemma 15.4.1 above and the fact that $\text{dist}(\hat{\mathbf{x}}, \mathbf{x}) \leq \sqrt{2} \|\hat{\mathbf{x}}\hat{\mathbf{x}}^* - \mathbf{x}\mathbf{x}^*\| / \|\mathbf{x}\|_{\ell_2}$.

15.4.2 Proof of stability of the WF initialization (Proof of first part of Theorem 12.3.2)

We begin by stating two lemmas from [226].

Lemma 15.4.2 [Matrix Gaussian Series]. Consider a finite sequence $\mathbf{A}_r \in \mathbb{R}^{n \times n}$ of fixed self-adjoint matrices, and let $\{g_r\}_{r=1}^m$ be a finite sequence of independent standard normal random variables. Compute the variance parameter

$$\Delta^2 := \left\| \sum_{r=1}^m \mathbf{A}_r^2 \right\|. \quad (15.4.1)$$

Then for all $t \geq 0$,

$$\mathbb{P} \left\{ \lambda_{\max} \left(\sum_{r=1}^m g_r \mathbf{A}_r \right) \geq t \right\} \leq n \cdot e^{-t^2/(2\Delta^2)}. \quad (15.4.2)$$

In particular,

$$\mathbb{P} \left\{ \left\| \left(\sum_{r=1}^m g_r \mathbf{A}_r \right) \right\| \geq t \right\} \leq 2n \cdot e^{-t^2/(2\Delta^2)}. \quad (15.4.3)$$

Lemma 15.4.3

Turning our attention to our proof note that

$$\frac{1}{m} \sum_{r=1}^m y_r \mathbf{a}_r \mathbf{a}_r^* = \frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{x}|^2 \mathbf{a}_r \mathbf{a}_r^* + \frac{1}{m} \sum_{r=1}^m w_r \mathbf{a}_r \mathbf{a}_r^*.$$

Therefore, the proof of the initialization result follows from combining the following lemma with the proofs of initialization for the noiseless case.

Lemma 15.4.4 *Assume $m \geq c(\delta)n \log n$ with c a fixed numerical constant depending only on δ . Then*

$$\left\| \frac{1}{m} \sum_{r=1}^m w_r \mathbf{a}_r \mathbf{a}_r^* \right\| \leq \delta \cdot \sigma, \quad (15.4.4)$$

holds with probability at least $1 - 4ne^{-n} - \frac{1}{n}$.

Proof Note that since \mathbf{w} has random sign we have

$$\frac{1}{m} \sum_{r=1}^m w_r \mathbf{a}_r \mathbf{a}_r^* = \frac{\sigma}{m} \sum_{r=1}^m g_r \mathbf{a}_r \mathbf{a}_r^*,$$

where $\{g_r\}_{r=1}^m$ are i.i.d. standard normal random variables that take the value ± 1 with equal probability. Setting $\mathbf{A}_r = \frac{\sigma}{m} \mathbf{a}_r \mathbf{a}_r^*$ we now turn our attention to bounding the variance parameter of Lemma 15.4.2. To this aim note that with probability at least $1 - 4ne^{-n}$

$$\sum_{r=1}^m \mathbf{A}_r^2 = \frac{\sigma^2}{m^2} \sum_{r=1}^m \|\mathbf{a}_r\|_{\ell_2}^2 \mathbf{a}_r \mathbf{a}_r^* \leq \frac{3n\sigma^2}{m^2} \sum_{r=1}^m \mathbf{a}_r \mathbf{a}_r^* \leq \frac{4n}{m} \sigma^2 \mathbf{I}. \quad (15.4.5)$$

Here, we have used the fact that since $\|\mathbf{a}_r\|_{\ell_2}^2$ is a Chi-squared random variable with probability at least $1 - 3ne^{-n}$, we have $\|\mathbf{a}_r\|_{\ell_2}^2 \leq 3n$ for $r = 1, 2, \dots, m$ and with probability at least $1 - 2e^{-n}$, $(\sum_{r=1}^m \mathbf{a}_r \mathbf{a}_r^*)/m \leq 4/3\mathbf{I}$. Thus, we can use $\Delta^2 = \frac{4n}{m} \sigma^2$ in Lemma 15.4.2 to arrive at

$$\mathbb{P} \left\{ \left\| \frac{1}{m} \sum_{r=1}^m w_r \mathbf{a}_r \mathbf{a}_r^* \right\| \geq 2\sigma t \sqrt{\frac{n}{m}} \right\} \leq 2n \cdot e^{-t^2/2}.$$

Using $t = \sqrt{2 \log n}$ completes the proof. ■

15.4.3 Proof of stability of the WF iteration updates (Proof of second part of Theorem 12.3.2)

In the previous Section we established that $\text{dist}(\mathbf{z}_0, \mathbf{x})$ is sufficiently small. This implies that \mathbf{z}_0 is either close to \mathbf{x} or $-\mathbf{x}$, without loss of generality we assume that \mathbf{z}_0 is close to \mathbf{x} . Using our SNR assumption together with the result of the previous section with the fact that $\hat{\mathbf{x}}$ is close to \mathbf{x} (proven in Theorem 12.3.1) we have

$$\|\mathbf{z}_0 - \hat{\mathbf{x}}\|_{\ell_2} \leq \frac{1}{8} \|\mathbf{x}\|_{\ell_2} + \sigma \leq \frac{1}{20} \|\mathbf{x}\|_{\ell_2}. \quad (15.4.6)$$

Therefore, it is sufficient to show (12.3.4) holds for the set $\{\mathbf{z} : \|\mathbf{z} - \hat{\mathbf{x}}\|_{\ell_2} \leq \frac{1}{20} \|\mathbf{x}\|_{\ell_2}\}$. Similar to the noiseless case we note that the algorithm is invariant to the norm of \mathbf{x} and therefore without loss of generality we shall assume $\|\mathbf{x}\|_{\ell_2} = 1$.

15.4.3.1 General convergence analysis

Define

$$E_{\mathbf{x}}(\epsilon) = \{\mathbf{z} : \text{dist}(\mathbf{z}, \mathbf{x}) \leq \epsilon\} \quad \text{and} \quad E_{\hat{\mathbf{x}}}(\epsilon) = \{\mathbf{z} : \text{dist}(\mathbf{z}, \hat{\mathbf{x}}) \leq \epsilon\}.$$

Using our SNR assumption together with the result of the previous section together with the fact that $\hat{\mathbf{x}}$ is close to \mathbf{x} (proven in Theorem 12.3.1), for any \mathbf{z} obeying $\text{dist}(\mathbf{z}, \hat{\mathbf{x}}) \leq \epsilon/2$ we have $\text{dist}(\mathbf{z}, \mathbf{x}) \leq \epsilon$ which implies $E_{\hat{\mathbf{x}}}(\epsilon/2) \subset E_{\mathbf{x}}(\epsilon)$.

We will assume that the function f satisfies a regularity condition on $E_{\mathbf{x}}(\epsilon)$, which essentially states that the Hessian of the function is well behaved.

Condition 15.4.5 (Hessian regularity condition) *We say that the function f satisfies the Hessian regularity condition or HRC(α, β, ϵ) iff for all vectors $\mathbf{z} \in E_{\mathbf{x}}(\epsilon)$ we have*

$$\alpha \mathbf{I} \leq \nabla^2 f(\mathbf{z}) \leq \beta \mathbf{I}. \quad (15.4.7)$$

In the lemma below we show that as long as the Hessian regularity condition holds on $E_{\mathbf{x}}(\epsilon)$ then Wirtinger Flow starting from an initial solution in $E_{\hat{\mathbf{x}}}(\epsilon/2)$ converges

to a global optimizer at a geometric rate. Subsequent sections shall establish that this property holds.

Lemma 15.4.6 *Assume that f obeys $HRC(\alpha, \beta, \epsilon)$ for all $\mathbf{z} \in E_{\mathbf{x}}(\epsilon)$. Furthermore, suppose $\mathbf{z}_0 \in E_{\hat{\mathbf{x}}}(\epsilon/2)$, and assume $0 < \mu \leq \frac{2}{\alpha+\beta}$. Consider the following update*

$$\mathbf{z}_{\tau+1} = \mathbf{z}_\tau - \mu \nabla f(\mathbf{z}_\tau).$$

Then for all τ we have $\mathbf{z}_\tau \in E_{\hat{\mathbf{x}}}(\epsilon/2)$ and

$$\text{dist}^2(\mathbf{z}_\tau, \hat{\mathbf{x}}) \leq \left(1 - \frac{2\alpha\beta\mu}{\alpha + \beta}\right)^\tau \text{dist}^2(\mathbf{z}_0, \hat{\mathbf{x}}).$$

Proof The proof is a spin on standard results in the optimization literature, e.g. see Theorem 2.1.15 of [186]. To prove this result we first prove the following lemma.

Lemma 15.4.7 *For all $\mathbf{z}, \mathbf{y} \in E_{\mathbf{x}}(\epsilon)$ we have*

$$\langle \nabla f(\mathbf{z}) - \nabla f(\mathbf{y}), \mathbf{z} - \mathbf{y} \rangle \geq \frac{\alpha\beta}{\alpha + \beta} \|\mathbf{z} - \mathbf{y}\|_{\ell_2}^2 + \frac{1}{\alpha + \beta} \|\nabla f(\mathbf{z}) - \nabla f(\mathbf{y})\|_{\ell_2}^2$$

Proof Denote $h(\mathbf{z}) = f(\mathbf{z}) - \frac{1}{2}\alpha \|\mathbf{z}\|_{\ell_2}^2$. Then $\nabla h(\mathbf{z}) = \nabla f(\mathbf{z}) - \alpha \mathbf{z}$ and $\nabla^2 h(\mathbf{z}) = \nabla^2 f(\mathbf{z}) - \alpha \mathbf{I}$. Thus,

$$\forall \mathbf{z} \in E_{\mathbf{x}}(\epsilon) \quad \nabla^2 h(\mathbf{z}) \leq (\beta - \alpha) \mathbf{I}$$

It is easy to see that this is equivalent to

$$\forall \mathbf{z}, \mathbf{y} \in E_{\mathbf{x}}(\epsilon), \quad \langle \nabla h(\mathbf{z}) - \nabla h(\mathbf{y}), \mathbf{z} - \mathbf{y} \rangle \geq \frac{1}{\beta - \alpha} \|\nabla h(\mathbf{z}) - \nabla h(\mathbf{y})\|_{\ell_2}^2,$$

which concludes the proof. ■

We now prove that $\mathbf{z}_\tau \in E_{\hat{\mathbf{x}}}(\epsilon/2)$ for all τ by induction. Note that using $\mathbf{z}_0 \in E_{\hat{\mathbf{x}}}(\epsilon/2)$

by (15.4.6) and assume that $\mathbf{z}_\tau \in E_{\hat{\mathbf{x}}}(\epsilon/2)$. We have

$$\begin{aligned}\|\mathbf{z}_{\tau+1} - \hat{\mathbf{x}}\|_{\ell_2}^2 &= \|\mathbf{z}_\tau - \hat{\mathbf{x}} - \mu \nabla f(\mathbf{z}_\tau)\|_{\ell_2}^2 \\ &= \|\mathbf{z}_\tau - \hat{\mathbf{x}}\|_{\ell_2}^2 - 2\mu \langle \nabla f(\mathbf{z}_\tau), \mathbf{z}_\tau - \hat{\mathbf{x}} \rangle + \mu^2 \|\nabla f(\mathbf{z}_\tau)\|_{\ell_2}^2\end{aligned}\quad (15.4.8)$$

Note that since $\hat{\mathbf{x}}, \mathbf{z}_\tau \in E_{\hat{\mathbf{x}}}(\epsilon/2) \subset E_{\hat{\mathbf{x}}}(\epsilon)$ then by Lemma 15.4.7 with $\mathbf{z} = \mathbf{z}_\tau$ and $\mathbf{y} = \hat{\mathbf{x}}$ we have

$$\langle \nabla f(\mathbf{z}_\tau) - \nabla f(\hat{\mathbf{x}}), \mathbf{z}_\tau - \hat{\mathbf{x}} \rangle \geq \frac{\alpha\beta}{\alpha+\beta} \|\mathbf{z}_\tau - \hat{\mathbf{x}}\|_{\ell_2}^2 + \frac{1}{\alpha+\beta} \|\nabla f(\mathbf{z}_\tau) - \nabla f(\hat{\mathbf{x}})\|_{\ell_2}^2$$

Also by optimality of $\hat{\mathbf{x}}$, $\nabla f(\hat{\mathbf{x}}) = 0$ and therefore we have

$$\langle \nabla f(\mathbf{z}_\tau), \mathbf{z}_\tau - \hat{\mathbf{x}} \rangle \geq \frac{\alpha\beta}{\alpha+\beta} \|\mathbf{z}_\tau - \hat{\mathbf{x}}\|_{\ell_2}^2 + \frac{1}{\alpha+\beta} \|\nabla f(\mathbf{z}_\tau)\|_{\ell_2}^2$$

Now we use the latter inequality to bound the second term in (15.4.8) and thus we have

$$\begin{aligned}\|\mathbf{z}_{\tau+1} - \hat{\mathbf{x}}\|_{\ell_2}^2 &= \|\mathbf{z}_\tau - \hat{\mathbf{x}}\|_{\ell_2}^2 - 2\mu \langle \nabla f(\mathbf{z}_\tau), \mathbf{z}_\tau - \hat{\mathbf{x}} \rangle + \mu^2 \|\nabla f(\mathbf{z}_\tau)\|_{\ell_2}^2 \\ &\leq \|\mathbf{z}_\tau - \hat{\mathbf{x}}\|_{\ell_2}^2 - 2\mu \left(\frac{\alpha\beta}{\alpha+\beta} \|\mathbf{z}_\tau - \hat{\mathbf{x}}\|_{\ell_2}^2 + \frac{1}{\alpha+\beta} \|\nabla f(\mathbf{z}_\tau)\|_{\ell_2}^2 \right) + \mu^2 \|\nabla f(\mathbf{z}_\tau)\|_{\ell_2}^2 \\ &= \left(1 - \frac{2\mu\alpha\beta}{\alpha+\beta}\right) \|\mathbf{z}_\tau - \hat{\mathbf{x}}\|_{\ell_2}^2 + \mu \left(\mu - \frac{2}{\alpha+\beta}\right) \|\nabla f(\mathbf{z}_\tau)\|_{\ell_2}^2 \\ &\leq \left(1 - \frac{2\mu\alpha\beta}{\alpha+\beta}\right) \|\mathbf{z}_\tau - \hat{\mathbf{x}}\|_{\ell_2}^2\end{aligned}$$

where the last line follows from the fact that $\mu \leq \frac{2}{\alpha+\beta}$. Since $\mathbf{z}_\tau \in E_{\hat{\mathbf{x}}}(\epsilon/2)$ it follows that $\mathbf{z}_{\tau+1} \in E_{\hat{\mathbf{x}}}(\epsilon/2)$ and by induction we have

$$\|\mathbf{z}_\tau - \hat{\mathbf{x}}\|_{\ell_2}^2 \leq \left(1 - \frac{2\mu\alpha\beta}{\alpha+\beta}\right)^\tau \|\mathbf{z}_0 - \hat{\mathbf{x}}\|_{\ell_2}^2,$$

concluding the proof. ■

15.4.3.2 Proof the Hessian regularity condition

In this section we establish the Hessian regularity condition. We shall prove the result for $\alpha = 0.9$, $\beta = 6n$, and $\epsilon = 1/10$. It is easy to calculate the gradient

$$\nabla f(\mathbf{z}) = \frac{1}{m} \sum_{r=1}^m (|\mathbf{a}_r^* \mathbf{z}|^2 - y_r) (\mathbf{a}_r^* \mathbf{z}) \mathbf{a}_r.$$

Similarly, the Hessian is given by

$$\nabla^2 f(\mathbf{z}) = \frac{1}{m} \sum_{r=1}^m (3|\mathbf{a}_r^* \mathbf{z}|^2 - y_r) \mathbf{a}_r \mathbf{a}_r^* = \frac{1}{m} \sum_{r=1}^m (3|\mathbf{a}_r^* \mathbf{z}|^2 - |\mathbf{a}_r^* \mathbf{x}|^2 - w_r) \mathbf{a}_r \mathbf{a}_r^*.$$

Note that by Lemma 15.4.4 and the SNR condition with high probability we have

$$\left\| \nabla^2 f(\mathbf{z}) - \frac{1}{m} \sum_{r=1}^m (3|\mathbf{a}_r^* \mathbf{z}|^2 - |\mathbf{a}_r^* \mathbf{x}|^2) \mathbf{a}_r \mathbf{a}_r^* \right\| \leq \delta_1,$$

where δ_1 is a sufficiently small numerical constant. Therefore, it suffices to show that that for all $\mathbf{z} \in E_{\mathbf{x}}(\epsilon)$

$$\alpha' \mathbf{I} \leq \frac{1}{m} \sum_{r=1}^m (3|\mathbf{a}_r^* \mathbf{z}|^2 - |\mathbf{a}_r^* \mathbf{x}|^2) \mathbf{a}_r \mathbf{a}_r^* \leq \beta' \mathbf{I}, \quad (15.4.9)$$

with $\alpha' = \alpha + \delta_1$ and $\beta' = \beta - \delta_1$. Before we prove (15.4.9) we start by stating some useful supporting lemmas and then we move on to prove the lower and upper bounds.

15.4.3.3 Supporting lemmas

Before providing the details of (15.4.9), we introduce two useful supporting lemmas.

Lemma 15.4.8 *Suppose $\{\mathbf{a}_r\}_{r=1}^m$, $m \geq c_0 \cdot n$ with c_0 a fixed numerical constant. Then with probability at least $1 - 2e^{-\gamma n}$ (γ is a numerical constant), we have*

$$(1 - \delta) \mathbf{I} \leq \frac{1}{m} \sum_{r=1}^m \mathbf{a}_r \mathbf{a}_r^* \leq (1 + \delta) \mathbf{I}. \quad (15.4.10)$$

Furthermore, if $m \geq c_0 \cdot n \log n$ with c_0 a fixed numerical constant. Then with probability at least $1 - 10e^{-\gamma n} - 8/n^2$ (γ is a numerical constant), we have

$$(1 - \delta)(\mathbf{I} + 2\mathbf{x}\mathbf{x}^*) \leq \frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{x}|^2 \mathbf{a}_r \mathbf{a}_r^* \leq (1 + \delta)(\mathbf{I} + 2\mathbf{x}\mathbf{x}^*), \quad (15.4.11)$$

where δ is a sufficiently small positive numerical constant.

Proof The inequality (15.4.10) essentially follows from the non-asymptotic bound for Wishart matrices as in the proof of Lemma 15.3.8. The inequality (15.4.11) follows from the real-valued version of 15.3.4. ■

We also make use of the following lemma which was proven by Bentkus in [36]:

Lemma 15.4.9 ([36]) Suppose X_r 's are i.i.d. random real variables for $r = 1, \dots, m$. Moreover, assume that $X_r \leq b$ for some nonrandom $b > 0$, and $\mathbb{E} X_r^2 = s^2$. Then

$$\mathbb{P}(X_1 + \dots + X_m \geq x) \leq \min(\exp(-\frac{x^2}{2\sigma^2}), c_0(1 - \Phi(\frac{x}{\sigma})))$$

for some $c_0 \leq 25$. Here $\sigma^2 = m \max(b^2, s^2)$.

15.4.3.4 Uniform upper bound on the Hessian

In this section, we establish that for any $\mathbf{z} \in E_{\mathbf{x}}(\epsilon)$, with $\epsilon = 1/10$ we have

$$\frac{1}{m} \sum_{r=1}^m (3|\mathbf{a}_r^* \mathbf{z}|^2 - |\mathbf{a}_r^* \mathbf{x}|^2) \mathbf{a}_r \mathbf{a}_r^* \leq 5.9n\mathbf{I}.$$

It suffices to prove that for any \mathbf{z} and \mathbf{y} satisfying $\|\mathbf{z}\|_{\ell_2} = \|\mathbf{y}\|_{\ell_2} = 1$, we have

$$\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{z}|^2 |\mathbf{a}_r^* \mathbf{y}|^2 \leq \frac{5.9}{3}n.$$

However by using Lemma 15.4.8 and the fact that $\|\mathbf{a}_r\|_{\ell_2} \leq 1.5n$ with high probability have

$$\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{z}|^2 |\mathbf{a}_r^* \mathbf{y}|^2 \leq \frac{1.5n}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{z}|^2 \leq \frac{5.9}{3}n,$$

concluding the proof.

15.4.3.5 Uniform lower bound on the Hessian

As to the lower bound, it suffices to prove that for all $\mathbf{z} \in E_{\mathbf{x}}(\epsilon)$, we have

$$\frac{1}{m} \sum_{r=1}^m (3|\mathbf{a}_r^* \mathbf{z}|^2 - |\mathbf{a}_r^* \mathbf{x}|^2) \mathbf{a}_r \mathbf{a}_r^* \geq \alpha' \mathbf{I}. \quad (15.4.12)$$

It follows from Lemma 15.4.8 that with high probability

$$\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{x}|^2 \mathbf{a}_r \mathbf{a}_r^* \leq (1 + \delta)(\mathbf{I} + 2\mathbf{x}\mathbf{x}^*),$$

provided $m \geq c_0 n \cdot \log n$ with c_0 a fixed numerical constant. (15.4.12) follows from combining the latter with the lemma below

Lemma 15.4.10 *Assume $\mathbf{z} \in \mathbb{R}^n$ is a fixed vector and \mathbf{u} and \mathbf{v} are two vectors with unit Euclidean norm, such that \mathbf{z} , \mathbf{u} and \mathbf{v} are pairwise orthogonal. Let s_1 and s_2 satisfy $s_1^2 + s_2^2 \leq \frac{1}{100}$, and t_1 , t_2 and t_3 satisfy $t_1^2 + t_2^2 + t_3^2 = 1$. Then, using $m \geq c_0 \cdot n \log n$ with c_0 a fixed numerical constant with probability at least $1 - \exp(-\gamma m)$ (γ is positive numerical constant)*

$$\frac{1}{m} \sum_{r=1}^m 3|\mathbf{a}_r^* \mathbf{z}|^2 |\mathbf{a}_r^* \mathbf{y}|^2 \geq \alpha' + (1 + \delta)(1 + 2|\mathbf{x}^* \mathbf{y}|^2) \quad (15.4.13)$$

holds for $\mathbf{z} = (1 + s_1)\mathbf{x} + s_2\mathbf{u}$ and $\mathbf{y} = t_1\mathbf{x} + t_2\mathbf{u} + t_3\mathbf{v}$.

Proof We shall prove this result for fixed vectors \mathbf{x} , \mathbf{u} and \mathbf{v} which are independent of the measurements. The Proof then follows by applying a covering argument (we skip the details). Rewriting (15.4.13) it suffices to prove

$$\frac{1}{m} \sum_{r=1}^m 3|\mathbf{a}_r^*((1 + s_1)\mathbf{x} + s_2\mathbf{u})|^2 |\mathbf{a}_r^*(t_1\mathbf{x} + t_2\mathbf{u} + t_3\mathbf{v})|^2 \geq \alpha' + (1 + \delta)(1 + 2t_1^2).$$

That is,

$$\frac{1}{m} \sum_{r=1}^m 3|(1 + s_1)Z_{r1} + s_2Z_{r2}|^2 |t_1Z_{r1} + t_2Z_{r2} + t_3Z_{r3}|^2 \geq \alpha' + (1 + \delta)(1 + t_1^2) = (1 + \alpha' + \delta) + 2(1 + \delta)t_1^2.$$

Here Z_{r1}, Z_{r2}, Z_{r3} are independent standard normal random variables.

We shall focus on the random variable

$$|(1 + s_1)Z_1 + s_2Z_2|^2 |t_1Z_1 + t_2Z_2 + t_3Z_3|^2.$$

We note that even though this random variable has a heavy right tail, its left tail is sub-Gaussian. Therefore, we can apply one-sided Bernstein-type inequalities such as Lemma 15.4.9 from [36]. To this end we will calculate the mean and upper bound the variance of this random variable. That is calculate

$$\mathbb{E}(|(1 + s_1)Z_1 + s_2Z_2|^2 |t_1Z_1 + t_2Z_2 + t_3Z_3|^2)$$

and give upper bounds for

$$\mathbb{E}(|(1 + s_1)Z_1 + s_2Z_2|^4 |t_1Z_1 + t_2Z_2 + t_3Z_3|^4).$$

15.4.3.5.1 Calculating the mean $\mathbb{E}(|(1 + s_1)Z_1 + s_2Z_2|^2 |t_1Z_1 + t_2Z_2 + t_3Z_3|^2)$

Straightforward calculations yield

$$\begin{aligned} & \mathbb{E}(|Z_1 + t_1Z_2|^2 |(t_2Z_1 + t_3Z_2 + t_4Z_3)|^2) \\ &= \mathbb{E}\left((1 + s_1)^2 t_1^2 Z_1^4 + ((1 + s_1)^2 t_2^2 + s_2^2 t_1^2 + 4(1 + s_1)s_2 t_1 t_2) Z_1^2 Z_2^2 + (1 + s_1)^2 t_3^2 Z_2^2 z_3^2 + s_2^2 t_2^2 Z_2^4 + s_2^2 t_3^2 Z_2^2 Z_3^2\right) \\ &= 3(1 + s_1)^2 t_1^2 + (1 + s_1)^2 t_2^2 + s_2^2 t_1^2 + 4(1 + s_1)s_2 t_1 t_2 + (1 + s_1)^2 t_3^2 + 3s_2^2 t_2^2 + s_2^2 t_3^2 \\ &= (3(1 + s_1)^2 + s_2^2)t_1^2 + ((1 + s_1)^2 + 3s_2^2)t_2^2 + 4(1 + s_1)s_2 t_1 t_2 + ((1 + s_1)^2 + s_2^2)t_3^2 := \Delta. \end{aligned}$$

Notice that by the assumptions $s_1^2 + s_2^2 \leq \frac{1}{100}$ and $t_1^2 + t_2^2 + t_3^2 = 1$, we have

$$\Delta = (2(1+s_1)^2+s_2^2)t_1^2+((1+s_1)^2-s_2^2)t_2^2+((1+s_1)t_1+2s_2t_2)^2+((1+s_1)^2+s_2^2)t_3^2 \geq \frac{81}{50}t_1^2+\frac{81}{100}(t_2^2+t_3^2).$$

and

$$\Delta \leq (3(1 + s_1)^2 + 3s_2^2)t_1^2 + (3(1 + s_1)^2 + 3s_2^2)t_2^2 + ((1 + s_1)^2 + s_2^2)t_3^2 \leq \frac{363}{100}.$$

15.4.3.5.2 Upper bound on the variance $\mathbb{E}(|(1+s_1)Z_1 + s_2Z_2|^4 | t_1Z_1 + t_2Z_2 + t_3Z_3|^4)$

By Cauchy-Schwarz, we have

$$\begin{aligned}\mathbb{E}(|(1+s_1)Z_1 + s_2Z_2|^4 | t_1Z_1 + t_2Z_2 + t_3Z_3|^4) &\leq \frac{1}{2} \mathbb{E}(|(1+s_1)Z_1 + s_2Z_2|^8 + |t_1Z_1 + t_2Z_2 + t_3Z_3|^8) \\ &= \frac{105}{2} \left(((1+s_1)^2 + s_2^2)^4 + 1 \right) \leq C_0\end{aligned}$$

where C_0 is a positive numerical constant.

Now we will use Lemma 15.4.9 to prove the proposition. Let

$$X_r = \Delta - |(1+s_1)Z_1 + s_2Z_2|^2 |t_1Z_1 + t_2Z_2 + t_3Z_3|^2.$$

Then $\mathbb{E}(X_r) = 0$, $X_r \leq \Delta \leq \frac{363}{100}$, and $\mathbb{E}(X_r^2) \leq C_0$. By Lemma 15.4.9, we have

$$\mathbb{P}(X_1 + \dots + X_m \geq \frac{m}{10}) \leq \exp(-\frac{m}{200C_0^2}) := \exp(-\gamma_0 m),$$

where γ_0 is a numerical constant. Therefore, with probability at least $1 - \exp(-\gamma_0 m)$, we have

$$X_1 + \dots + X_m \leq \frac{m}{10},$$

which is

$$\frac{1}{m} \sum_{r=1}^m |(1+s_1)Z_{r1} + s_2Z_{r2}|^2 |t_1Z_{r1} + t_2Z_{r2} + t_3Z_{r3}|^2 \geq \Delta - \frac{1}{10} \geq \frac{81}{100}t_1^2 + \frac{71}{100}.$$

Our proof is complete by picking $\alpha' = 0.915$ and $\delta = 0.215$ which obey $\frac{1+\alpha'+\delta}{3} \leq \frac{71}{100}$ and $\frac{2(1+\delta)}{3} \leq \frac{81}{100}$. ■

15.5 Proofs for the error reduction algorithm

In this section we prove the results about the error reduction algorithm. From the initialization result of Theorem 12.2.4 for the real Gaussian model we have

$$\min(\|z_0 - \mathbf{x}\|_{\ell_2}, \|z_0 + \mathbf{x}\|_{\ell_2}) \leq \frac{\sqrt{m} - \sqrt{n}}{\sqrt{m} + \sqrt{n}} \frac{1}{\sqrt{32n}} \min_r |\mathbf{a}_r^* \mathbf{x}|.$$

Therefore with probability at least $1 - e^{-n}$ we have

$$\min(\|\mathbf{v}_0 - \mathbf{A}\mathbf{x}\|_{\ell_2}, \|\mathbf{v}_0 + \mathbf{A}\mathbf{x}\|_{\ell_2}) \leq \|\mathbf{A}\| \min(\|z_0 - \mathbf{x}\|_{\ell_2}, \|z_0 + \mathbf{x}\|_{\ell_2}) < \frac{\sqrt{m} - \sqrt{n}}{\sqrt{8n}} \min_r |\mathbf{a}_r^* \mathbf{x}|.$$

In the above we have used the fact that with probability at least $1 - e^{-n}$, $\|\mathbf{A}\| \leq 2(\sqrt{m} + \sqrt{n})$. This concludes the proof of (13.2.1).

To prove the convergence result note that when (13.2.1) holds, for $\mathbf{z}_0 = (\mathbf{A}^* \mathbf{A})^{-1} \mathbf{A}^* \mathbf{v}_0$ we have

$$\begin{aligned} \min(\|z_0 - \mathbf{x}\|_{\ell_2}, \|z_0 + \mathbf{x}\|_{\ell_2}) &\leq \|(\mathbf{A}^* \mathbf{A})^{-1} \mathbf{A}^*\| \min(\|\mathbf{v}_0 - \mathbf{A}\mathbf{x}\|_{\ell_2}, \|\mathbf{v}_0 + \mathbf{A}\mathbf{x}\|_{\ell_2}), \\ &< \frac{1}{\sqrt{2n}} \min_r |\mathbf{a}_r^* \mathbf{x}|. \end{aligned}$$

In the above we have used the fact that with probability at least $1 - e^{-n}$, for $m \geq 9n$ we have $\sigma_{\min}(\mathbf{A}) \geq \frac{1}{2}(\sqrt{m} - \sqrt{n})$.

Therefore,

$$\begin{aligned} \min(\|\mathbf{v}_0 - \mathbf{A}\mathbf{x}\|_{\ell_\infty}, \|\mathbf{v}_0 + \mathbf{A}\mathbf{x}\|_{\ell_\infty}) &= \min(\|\mathbf{A}\mathbf{z}_0 - \mathbf{A}\mathbf{x}\|_{\ell_\infty}, \|\mathbf{A}\mathbf{z}_0 + \mathbf{A}\mathbf{x}\|_{\ell_\infty}) \\ &\leq \max_r \|\mathbf{a}_r\|_{\ell_2} \min(\|z_0 - \mathbf{x}\|_{\ell_2}, \|z_0 + \mathbf{x}\|_{\ell_2}) \\ &< \min_r |\mathbf{a}_r^* \mathbf{x}|. \end{aligned}$$

Define

$$\begin{aligned} E &= \{\mathbf{v} \in \mathbb{R}^m : \min(\|\mathbf{v} - \mathbf{Ax}\|_{\ell_2}, \|\mathbf{v} + \mathbf{Ax}\|_{\ell_2}) < \frac{\sqrt{m} - \sqrt{n}}{\sqrt{8n}} \min_r(|\mathbf{a}_r^* \mathbf{x}|)\} \\ F &= \{\mathbf{v} \in \mathbb{R}^m : \min(\|\mathbf{v} - \mathbf{Ax}\|_{\ell_\infty}, \|\mathbf{v} + \mathbf{Ax}\|_{\ell_\infty}) < \min_r(|\mathbf{a}_r^* \mathbf{x}|)\}. \end{aligned} \quad (15.5.1)$$

Based on the argument above $E \subset F$. This implies that if $\mathbf{v} \in E$ for all r we have

$$|v_r| - b_r \leq \min(|v_r - \mathbf{a}_r^* \mathbf{x}|, |v_r + \mathbf{a}_r^* \mathbf{x}|) < |\mathbf{a}_r^* \mathbf{x}|.$$

Note that the function $h(v) = \frac{1}{2}(v - b)^2$ for $b > 0$ is convex and differentiable for $|v| - b < b$ with $h''(v) = 1$. Thus for any $\mathbf{v} \in E$ we have $\nabla^2 f(\mathbf{v}) = \mathbf{I}$. Therefore, it follows from Lemma 15.5.1 below that with high probability

$$\begin{aligned} \text{dist}(\mathbf{v}_\tau, \mathbf{Ax}) &\leq (1 - \mu)^{\tau/2} \text{dist}(\mathbf{v}_0, \mathbf{Ax}) \\ &< (1 - \mu)^{\tau/2} \frac{\sqrt{m} - \sqrt{n}}{\sqrt{8n}} \min_r |\mathbf{a}_r^* \mathbf{x}| \\ &\leq \frac{\sqrt{m} - \sqrt{n}}{\sqrt{4\pi n}} (1 - \mu)^{\tau/2} \cdot \|\mathbf{x}\|_{\ell_2}, \end{aligned}$$

concluding the proof of (13.2.2) and the theorem. In the above lemma we have used the fact that the minimum of m independent standard normal random variables Z_1, Z_2, \dots, Z_m obeys $\min_r |Z_r| \leq \sqrt{2/\pi}$ with probability at least $1 - 2^{-n}$.

Lemma 15.5.1 *Let f be a convex function with continuous second order derivatives on a set F obeying*

$$\alpha \mathbf{I} \leq \nabla^2 f(\mathbf{v}) \leq \beta \mathbf{I} \quad \text{for all } \mathbf{v} \in F.$$

Furthermore, assume the set $E = \{\mathbf{v} \in \mathbb{R}^m : \|\mathbf{v} - \mathbf{u}\|_{\ell_2} \leq r\} \subset F$ where \mathbf{u} is the unique minimum of the function f on F . Also assume \mathcal{Q} is a convex subset of \mathbb{R}^n and $\mathbf{u} \in \mathcal{Q}$. Then for any $\mathbf{v}_0 \in E$ and $\mu \leq 1/\beta$ the update

$$\mathbf{v}_{\tau+1} = \mathbf{v}_\tau - \mu g_{\mathcal{Q}}(\mathbf{v}_\tau; \beta), \quad (15.5.2)$$

obeys

$$\|\mathbf{z}_\tau - \mathbf{u}\|_{\ell_2}^2 \leq (1 - \alpha\mu)^\tau \|\mathbf{z}_0 - \mathbf{u}\|_{\ell_2}^2.$$

Proof The proof follows by induction. Assume $\mathbf{v}_\tau \in E$. For any $\mathbf{v}_\tau \in E$ we have

$$\begin{aligned} \|\mathbf{v}_{\tau+1} - \mathbf{u}\|_{\ell_2}^2 &= \|\mathbf{v}_\tau - g_{\mathcal{Q}}(\mathbf{v}_\tau; L) - \mathbf{u}\|_{\ell_2}^2 \\ &= \|\mathbf{v}_\tau - \mathbf{u}\|_{\ell_2}^2 - 2\mu \langle g_{\mathcal{Q}}(\mathbf{v}_\tau; L), \mathbf{v}_\tau - \mathbf{u} \rangle + \mu^2 \|g_{\mathcal{Q}}(\mathbf{v}_\tau; L)\|_{\ell_2}^2 \\ &\leq (1 - \alpha\mu) \|\mathbf{v}_\tau - \mathbf{u}\|_{\ell_2}^2 + \mu(\mu - \frac{1}{\beta}) \|g_{\mathcal{Q}}(\mathbf{v}_\tau; L)\|_{\ell_2}^2 \\ &\leq (1 - \alpha\mu) \|\mathbf{v}_\tau - \mathbf{u}\|_{\ell_2}^2. \end{aligned}$$

In the first inequality we have used the lemma below. ■

Lemma 15.5.2 *Let f be a function obeying*

$$\alpha \mathbf{I} \preceq \nabla^f(\mathbf{v}) \preceq \beta \mathbf{I}.$$

Furthermore, assume $\gamma \geq \beta$. Then for any $\mathbf{v} \in \mathcal{Q}$

$$\langle g_{\mathcal{Q}}(\mathbf{v}, \gamma), \mathbf{v} - \mathbf{u} \rangle \geq \frac{1}{2\gamma} \|g_{\mathcal{Q}}(\mathbf{v}, \gamma)\|_{\ell_2}^2 + \frac{\alpha}{2} \|\mathbf{v} - \mathbf{u}\|_{\ell_2}^2.$$

Here, \mathbf{u} is the unique global minimum of f on the set \mathcal{Q} .

Proof The proof is standard e.g. see Corollary 2.3.2 of [186]. ■

15.5.1 convergence of the iterations of the error reduction

15.5.2 Proof of stability of the global optimum of the WF objective (Proof of Theorem 12.3.1)

Part III

Compressed Sensing, Denoising and Sparse Recovery with Coherent and Redundant Dictionaries

Chapter 16

Background

Compressed sensing is a signal acquisition technique that allows efficient reconstruction of a signal from under-determined linear measurements by exploiting sparsity or compressibility of the signal. By now classic theorems in this field [63, 87] show that it is possible to recover a signal from under-sampled random measurements as long as the signal is sparse in an ortho-basis. In this theory the required number of measurements is proportional to the sparsity of the signal, and therefore allows for far fewer measurements than traditional data acquisition techniques. The early papers on compressive sensing [63, 87] triggered a tremendous amount of literature both in terms of the applications and theory of compressive sensing. Despite all this progress in the last decade and more than 10,000 published papers in the field, some of the most fundamental mathematical questions related to compressed sensing are still not understood. For example, in most signal processing and data analysis applications the signal of interest is not sparse in an ortho-basis but in terms of an over-complete dictionary such as over-complete wavelets, low-pass Fourier and Gabor dictionaries. Indeed, recovery of a signal (that is sparse in an over-complete dictionary) was one of the main motivating examples in the original compressed sensing papers [63, 87]. However, there is little theory that explains why this is possible.

In a related problem, sparse recovery and estimation in over-complete dictionaries are often of interest in applications such as microscopy, astronomy, tomography,

computer vision, radar, and seismology. Conventional wisdom in sparse signal recovery postulates that to approximately recover a sparse signal \mathbf{x} from a system of under-determined linear equations of the form $\mathbf{f} = \mathbf{D}\mathbf{x}$ via ℓ_1 minimization, the columns of the matrix \mathbf{D} need to be *incoherent*, i.e. have small dot products. For example, this is the basis behind conditions such as the restricted isometry property [65] or the restricted eigenvalue condition [41]. However, most of the dictionaries we encounter in signal processing, neuroscience and computer vision, such as low-pass Fourier, wavelets, spherical harmonics, and over-complete Gabor, do not obey such properties. Nevertheless ℓ_1 minimization can be quite effective for these applications.

In this part of the dissertation we wish to address the problems discussed above. In Chapter 17 we provide some theory for compressive sensing with coherent and redundant dictionaries. Later in the same chapter we shall use this result to connect compressive sensing in over-complete dictionaries to denoising using an over-complete dictionary. In Chapter 18 we develop theory for sparse recovery with highly coherent dictionaries which encompasses certain classes of over-complete wavelets, Gabor and low pass Fourier matrices.¹ These results show that sparse recovery via ℓ_1 is effective in these dictionaries even though these dictionaries have maximum pair-wise column coherence very close to 1, i.e. they contain almost identical columns. This holds with the proviso that the sparse coefficients are not too clustered. This general theory, when applied to the special case of low pass Fourier, allows for less restrictive requirements compared with recent literature [54, 220]² with significantly simpler and shorter proofs. All of the results in this part are based on yet unpublished notes by the author.³

¹We emphasize that although our theory is widely applicable, in this dissertation we shall only provide the details for an over-complete low-pass Fourier matrix.

²Please see [40, 53, 213, 219] for some related literature.

³Some of these results (more specifically the results on compressive sensing and denoising with coherent and redundant dictionaries) date back to 2011. All of these results with greater detail will appear in future publications by the author.

Chapter 17

Compressive sensing and denoising with coherent and redundant dictionaries

17.1 Problem statement and preliminaries

In this section we shall discuss two problems involving coherent and redundant dictionaries.

17.1.1 Compressed sensing with coherent and redundant dictionaries

In most signal processing applications a signal of interest is not sparse in an orthonormal basis rather sparsity is expressed in terms of an *overcomplete* dictionary. This means that the signal $\mathbf{f} \in \mathbb{R}^n$ can be written in the form $\mathbf{f} = \mathbf{D}\mathbf{x}$ where $\mathbf{D} \in \mathbb{R}^{n \times N}$ (typically $n < N$) is the dictionary and $\mathbf{x} \in \mathbb{R}^N$ is the sparse coefficients. Now we wish to recover this signal from random linear measurements of the form

$$\mathbf{y} = \mathbf{A}\mathbf{f}.$$

Here $\mathbf{A} \in \mathbb{R}^{m \times n}$ (typically $m \ll n$) is the measurement matrix. We consider recovery via the following optimization problem.

$$\begin{aligned} \min_{\bar{\mathbf{x}} \in \mathbb{R}^N} \quad & \|\bar{\mathbf{x}}\|_{\ell_1} \\ \text{subject to} \quad & \mathbf{y} = \mathbf{A}\bar{\mathbf{x}}. \end{aligned} \tag{17.1.1}$$

After finding a solution $\hat{\mathbf{x}}$ (does not have to be unique) we estimate the signal by $\hat{\mathbf{f}} = \mathbf{D}\hat{\mathbf{x}}$. This is known as the synthesis problem. We are interested in knowing when the synthesis problem gives us the correct answer, that is, when $\hat{\mathbf{f}} = \mathbf{f}$.

17.1.2 Denosing with coherent and redundant dictionaries

In this problem we again have a signal $\mathbf{f} \in \mathbb{R}^n$ which is approximately sparse in an over-complete dictionary. That is, $\mathbf{f} = \mathbf{D}\mathbf{x}$ where $\mathbf{D} \in \mathbb{R}^{n \times N}$ and $\mathbf{x} \in \mathbb{R}^N$ is an approximately sparse signal. However, we do not get to see \mathbf{f} itself but a noisy version of f . In particular we assume that we have measurements of the form $\mathbf{y} = \mathbf{f} + \mathbf{w}$ where $\mathbf{w} \in \mathbb{R}^n$ denotes the corruption. Throughout this part we assume \mathbf{w} is a Gaussian random vector with entries $\mathcal{N}(0, \frac{\sigma^2}{n})$. Now given these noisy measurements \mathbf{y} we wish to estimate the signal \mathbf{f} . It is natural to use the following optimization scheme

$$\hat{\mathbf{f}} = \mathbf{D}\hat{\mathbf{x}} \quad \text{where} \quad \hat{\mathbf{x}} = \arg \min_{\bar{\mathbf{x}}} \quad \frac{1}{2} \|\mathbf{y} - \mathbf{D}\bar{\mathbf{x}}\|_{\ell_2}^2 + \lambda \|\bar{\mathbf{x}}\|_{\ell_1}. \tag{17.1.2}$$

We are interested in understanding when our estimate $\hat{\mathbf{f}}$ is a good approximation for the signal \mathbf{f} .

17.2 Theory

In this section we shall discuss our main theoretical results. Before we begin however we need some geometric definitions to proceed.

17.2.1 Some geometric definitions and assumptions

Given a dictionary $\mathbf{D} \in \mathbb{R}^{n \times N}$ (or $\mathbb{C}^{n \times N}$) we define the polytope \mathcal{D} as the symmetrized convex hull of columns of \mathbf{D} , henceforth denoted by $\text{symconv}(\mathbf{D})$. In the real case we define the symmetrized convex hull by

$$\text{symconv}(\mathbf{D}) = \text{conv}(\pm \mathbf{D}_1, \pm \mathbf{D}_2, \dots, \pm \mathbf{D}_N).$$

In the complex case, i.e. when $\mathbf{D} \in \mathbb{C}^{n \times N}$ the definition is

$$\text{symconv}(\mathbf{D}) = \text{conv}_{\phi_1, \dots, \phi_N \in [0, 2\pi)} (e^{i\phi_1} \mathbf{D}_1, e^{i\phi_2} \mathbf{D}_2, \dots, e^{i\phi_N} \mathbf{D}_N).$$

Throughout this chapter we assume that the signal \mathbf{f} lies on an outer face \mathcal{F} of the polytope \mathcal{D} after appropriate dilation. In this case, we will say that \mathbf{f} belong to the face \mathcal{F} .

Definition 17.2.1 *Given the outer face \mathcal{F} we define a subset of the columns of \mathbf{D} by its endpoints*

$$T = \{ k \mid \exists \epsilon_k \in \{\pm 1\} \text{ with } \epsilon_k \mathbf{D}_k \in \text{aff}(\mathcal{F}) \}.$$

We assume $|T| = s$ and use $\epsilon_1, \epsilon_2, \dots, \epsilon_s$ to denote the corresponding sign pattern. Without loss of generality we assume from here onwards that the columns indexed by T are the first s columns of \mathbf{D} . Furthermore, we use $\tilde{\mathbf{D}}_T$ to denote the signed version of the columns in T , i.e. $\tilde{\mathbf{D}}_T = [\epsilon_1 \mathbf{D}_1, \epsilon_2 \mathbf{D}_2, \dots, \epsilon_s \mathbf{D}_s]$.

In the complex case the definition becomes

$$T = \{ k \mid \exists \epsilon_k = e^{i\phi_k}, \phi_k \in [0, 2\pi) \text{ with } \epsilon_k \mathbf{D}_k \in \text{aff}(\mathcal{F}) \}.$$

Furthermore, in the complex case the definition of $\text{aff}(\mathcal{F})$ is

$$\text{aff}(\mathcal{F}) = \left\{ \sum_{k=1}^s \theta_k \epsilon_k \mathbf{D}_k \mid \sum_{k=1}^s \theta_k = 1, \text{ with } \theta_k \in \mathbb{R} \right\},$$

where $\epsilon_k \mathbf{D}_k$ are the endpoints of face \mathcal{F} . We need to clarify by what we mean by “face” in the complex case. Throughout, we use face \mathcal{F} to denote

$$\mathcal{F} = \left\{ \sum_{k=1}^s \theta_k \epsilon_k \mathbf{D}_k \mid \sum_{k=1}^s \theta_k = 1, \text{ with } \theta_k \in \mathbb{R} \text{ and } \theta_k \geq 0 \right\},$$

Next we define a new parameter that we believe has a very important role in compressed sensing over general polytopes.

Definition 17.2.2 (distance of a face with respect to a polytope) *Given an outer face \mathcal{F} of the polytope \mathcal{D} we define the distance of this face with respect to the polytope \mathcal{D} as the distance between the affine hull of face \mathcal{F} and the symmetric convex hull of the rest of the points.*

$$\text{dist}(\mathcal{F}, \mathcal{D}) = \text{dist}\left(\text{aff}(\mathcal{F}), \text{symconv}\left(\left[\mathbf{D}_{s+1}, \dots, \mathbf{D}_N\right]\right)\right). \quad (17.2.1)$$

Notice that for the case that \mathcal{D} is the ℓ_1 ball the definition yields the answer

$$\text{dist}(\mathcal{F}, \text{symconv}(\mathcal{I})) = \frac{1}{\sqrt{s}} \quad (17.2.2)$$

for any face \mathcal{F} of dimension s , which is the crucial parameter in classical compressed sensing results.

We say that $\boldsymbol{\nu}$ is a dual certificate for the face \mathcal{F} if

$$\begin{aligned} \tilde{\mathbf{D}}_T * \boldsymbol{\nu} &= \mathbf{1} \\ \|\mathbf{D}_{T^c}^* \boldsymbol{\nu}\|_{\ell_\infty} &= \alpha < 1. \end{aligned}$$

We end with the definition of the mean width.

Definition 17.2.3 *The mean width $M^*(\mathcal{D})$ of a symmetric convex body \mathcal{D} in \mathbb{R}^n is the expected value of the dual norm over the unit sphere,*

$$M^*(\mathcal{D}) = M(\mathcal{D}^o) = \int_{S^{n-1}} \|\mathbf{y}\|_{\mathcal{D}^o} d\sigma(\mathbf{y}) = \int_{S^{n-1}} \max_{\mathbf{z} \in \mathcal{D}} \langle \mathbf{y}, \mathbf{z} \rangle d\sigma(\mathbf{y}). \quad (17.2.3)$$

17.2.2 Theory for compressed sensing with coherent and redundant dictionaries

With all the auxiliary lemmas in place we are now ready to state our main sampling theorem.

Theorem 17.2.4 *Given any signal \mathbf{f} on an outer face \mathcal{F} with distance $\text{dist}(\mathcal{F}, \mathcal{D})$ with respect to a polytope with endpoints on the unit sphere then the synthesis procedure (17.1.1) recovers \mathbf{f} exactly, provided*

$$m > cn \left(\frac{M^*(\mathcal{D})}{\text{dist}(\mathcal{F}, \mathcal{D})} \right)^2. \quad (17.2.4)$$

where c is a global constant.

First, while we state the result in terms of Gaussian matrices a similar identity also holds for a large class of random matrices, including matrices with entries i.i.d. ± 1 . The reason is that the proof is based on calculating upper bounds on the intersection of the intersection of the row space of the sensing matrix \mathbf{A} with the polytope.¹ In the geometric functional analysis literature these quantities are well understood not only for Gaussian random matrices but for many other ensembles. Please see the proofs for further details.

Second, note that in the case where $\mathbf{D} = \mathbf{I}$, $M^*(\mathcal{D}) = c\sqrt{\frac{\log n}{n}}$ and $\text{dist}(\mathcal{F}, \mathcal{D}) = \frac{1}{\sqrt{s}}$ for a face \mathcal{F} of sparsity s . Therefore the above identity predicts $m > cs \log n$. So our general result reduces to more classical results when the dictionary is the identity matrix.

Our result may seem a bit pedagogical and not practical at this point. In particular, it is not clear how one would estimate the mean width and the distance quantity in the above result. First, when the columns of the dictionary \mathbf{D} have unit norm one can often show that $M^*(\mathcal{D}) \leq c\sqrt{\frac{\log N}{n}}$ for a fixed numerical constant c . Second, it is easy to provide a lower bound on the distance quantity by means of dual certificates of the face \mathcal{F} using the lemma below.

¹We note that in the complex case $\text{symconv}(\mathbf{D})$ is not technically a polytope. However, throughout we shall use the word polytope for the complex case as well by which we mean $\text{symconv}(\mathbf{D})$.

Lemma 17.2.5 (connection between the dual certificate and the distance)

We have

$$\text{dist}(\mathcal{F}, \mathcal{D}) = \max_{\nu \in \mathbb{R}^n : \tilde{\mathbf{D}}_T^* \nu = 1} \frac{1 - \|\mathbf{D}_{T^c}^* \nu\|_{\ell_\infty}}{\|\nu\|_{\ell_2}}.$$

Using the above lemma as well as the approximation on the mean width stated above the required number of samples is given by

$$m > c \log N \left(\frac{\|\nu\|_{\ell_2}}{1 - \|\mathbf{D}_{T^c}^* \nu\|_{\ell_\infty}} \right)^2 > cn \left(\frac{M^*(\mathcal{D})}{\frac{1 - \|\mathbf{D}_{T^c}^* \nu\|_{\ell_\infty}}{\|\nu\|_{\ell_2}}} \right)^2 \geq cn \left(\frac{M^*(\mathcal{D})}{\text{dist}(\mathcal{F}, \mathcal{D})} \right)^2.$$

Note that for dictionaries \mathbf{D} for which sparse recovery via ℓ_1 minimization is well understood we often can construct the dual certificate or an approximate version of it so that we can easily estimate its $\|\nu\|_{\ell_2}$ and the quantity $\|\mathbf{D}_{T^c}^* \nu\|_{\ell_\infty}$ (α). Please see Chapter 18 for an example.

17.2.3 Theory for denoising with coherent and redundant dictionaries

Consider,

$$\mathbf{x} = \arg \min_{\bar{\mathbf{x}} \in \mathbb{R}^N} \|\bar{\mathbf{x}}\|_{\ell_1} \quad \text{subject to} \quad \mathbf{f} = \mathbf{D}\bar{\mathbf{x}}.$$

We assume that T consists of the $|T|$ largest coefficients of \mathbf{x} in absolute value.

$$\mathbf{f} = \mathbf{f}_T + \mathbf{f}_{T^c} \quad \text{where} \quad \mathbf{f}_T = \mathbf{D}_T \mathbf{x}_T \in \mathcal{F} \quad \text{and} \quad \mathbf{f}_{T^c} = \mathbf{D}_{T^c} \mathbf{x}_{T^c}.$$

Theorem 17.2.6 (Denoising) *Using $\lambda = C \frac{2}{1-\alpha} \sqrt{\log N} \frac{\sigma}{\sqrt{n}}$ with $C \geq 1$ in (17.1.2) we have*

$$\|\hat{\mathbf{f}} - \mathbf{f}\|_{\ell_2}^2 \leq 34 \|\mathbf{f}_{T^c}\|_{\ell_2}^2 + 32 \frac{s}{n} \sigma^2 + 128 C^2 \frac{\sigma^2}{n} \frac{\log N}{\text{dist}(\mathcal{F}, \mathcal{D})^2}.$$

We note that for $\mathbf{D} = \mathbf{I}$, $\text{dist}(\mathcal{F}, \mathcal{D}) = \frac{1}{\sqrt{s}}$ and therefore the above reduces to well known results. One interesting aspect of this result is that both denoising and compressive sensing using redundant dictionaries depend inversely on our notion of distance. This

theorem connects compressed sensing to denoising. The reason is that the same factor appears in compressed sensing of over-complete dictionaries also appears in the ℓ_2/ℓ_2 denoising results above. That is the ℓ_2/ℓ_2 recovery result is inversely proportional to the distance quantity and so is the required number of measurements.

17.2.4 Comparison with some related literature

We briefly pause to compare our results with some recent literature on this topic. There are many papers written on the subject of compressed sensing and we can not possibly hope to review all of this. Rather we shall focus on only a subset of the results which are most related to our own. Another result which studies compressive sensing with coherent and redundant dictionaries is [51] which studies this problem when the transpose of the dictionary times the signal $\mathbf{D}^* \mathbf{f}$ is sparse. This problem is often referred to as the analysis problem. Please also see [183, 184, 201, 230] for a few followup papers and more discussions. The connection between denoising and the required number of measurements was first formalized in a series of conjectures [88, 89, 91] about the coincidence between the minimax risk of denoising and phase transitions of compressive measurements. As stated earlier the results of this chapter make progress towards these conjectures albeit up to unknown constants.² In 2014, Oymak and Hassibi [194] showed that the minimax risk of denoising using convex regularization can be precisely characterized using geometric properties of the convex cost function mainly via a quantity known as the statistical dimension (Please see [165] for a precise definition). Finally, in a fantastic paper [13] the authors show that the phase transition of the success of compressive sensing with convex regularizers, is also characterized precisely by the statistical dimension. This result when combined with that of [194] establishes the conjecture of [89] for Gaussian matrices and convex regularizers. In comparison our result is not precise in terms of the constant. Also, our result only applies to convex regularizers whose unit ball are the convex hull of a finite number of points. However, our results does have two advantages when compared with this more recent literature. First, our result (as we explained) also

²We note that the results mentioned in this chapter were established by the author in 2011 before these conjectures were formalized and before the papers subsequently discussed.

applies to other sensing ensembles. Second, the statistical dimension parameter is not easy to calculate in case of the synthesis problem, whereas we have explained how to calculate our distance notion via a dual certificate construction. That being said, the author believes these results are complementary in nature. Indeed, the author believes that the distance calculations discussed in this chapter can be viewed as an effective (albeit not tight in terms of the constant) way to bound the statistical dimension. Rigorously establishing this connection (and perhaps even improving upon it) is an interesting direction for future research.

Chapter 18

Sparse recovery with highly coherent dictionaries

In this chapter we shall provide a general proof methodology for sparse recovery in highly coherent dictionaries. We shall provide the details of the argument for the super-resolution problem (to be defined below).

18.1 Two models and their connection

In this section we shall explain two different models for the super-resolution problem and then connect these two cases to each other.

18.1.1 Continuous frequency model

Define

$$x_c(t) = \sum_{s \in S} x[s] \delta(t - s),$$

where $S \subset \mathbb{R}$ is a discrete set of time instances. Let $X_c(f)$ denote the Fourier transform of $x_c(t)$. We assume that we can observe $X_c(f)$ in the interval $[-\Omega, \Omega]$. Given this observation we would like to extrapolate the spectrum of $X_c(f)$ at the other

frequencies. For this purpose we solve the following convex program

$$\hat{x} = \arg \min_{\bar{x}} \|\bar{x}\|_{TV} \quad \text{subject to} \quad \bar{X}(f) = X_c(f) \text{ for } f \in [-\Omega, \Omega]. \quad (18.1.1)$$

To the extent of our knowledge this approach to extrapolation was first proposed by Arne Beurling in 1938¹ [37–39, 68] and later utilized by many authors [54, 86, 90, 152–154]. We are interested in understanding when (18.1.1) yields exact recovery (exact extrapolation). For this purpose we assume that the elements of S are well separated.

Definition 18.1.1 (minimum distance)

$$\Delta_c(S) = \min_{s \neq s' \in S} |s - s'|.$$

Definition 18.1.2 (Over Sampling Ratio (OSR))

$$OSR_c = 2\Omega\Delta_c(S).$$

18.1.2 Discrete frequency model

Define

$$x_d(t) = \sum_{s \in S} x[s]\delta(t - s),$$

where $S \subset [0, 1]$ is a discrete set of time instances. Let $X_d(f)$ denote the Fourier series of $x_d(t)$ defined by

$$X_d(f) = \int_{\tau_0}^{\tau_0+1} x_d(\tau) \cdot e^{-2\pi i f \tau} d\tau, \quad \text{for } f \in \mathbb{Z}.$$

We assume that we can observe $X_d(f)$ for $f = 0, \pm 1, \pm 2, \dots, \pm f_c$. Given these observations we would like to extrapolate the spectrum of $X_d(f)$ at the other frequencies.

¹We note however that when the coefficients are positive this approach first appeared (and was proven) more than a hundred years ago in the writings of Caratheodory [67] and was later rediscovered by multiple authors [92, 110, 112].

For this purpose we solve the following convex program

$$\hat{x} = \arg \min_{\bar{x}} \|\bar{x}\|_{TV} \quad \text{subject to} \quad \bar{X}(f) = X_d(f) \text{ for } f = 0, \pm 1, \pm 2, \dots, \pm f_c. \quad (18.1.2)$$

We are interested in understanding when (18.1.1) yields exact recovery (exact extrapolation). For this purpose we assume that the elements of S are well separated.

Definition 18.1.3 (minimum distance)

$$\Delta_d(S) = \min_{s \neq s' \in S} |s - s'|.$$

Here the distance is the wrap around distance, as in [54].

Definition 18.1.4 (Over Sampling Ratio (OSR))

$$OSR_d = 2f_c \Delta_d(S).$$

18.1.3 The connection

In this section we shall prove that exact recovery for the continuous model implies exact recovery for the descrete model. Using this connection if we are interested in establishing an exact recovery result for the discrete model it suffices to focus on the continuous model.

Theorem 18.1.5 *If (18.1.1) yields exact recovery for any signal with $OSR_c \geq \eta$, then (18.1.2) also yields exact recovery for any signal with $OSR_d \geq \eta$.*

18.2 Continuous Super-resolution via TV-minimization

Without any delay we shall explain our main result.

Theorem 18.2.1 *If $\Omega > 23.1$, then as long as*

$$OSR \geq 3.12$$

(18.1.1) yields exact recovery.

We note that the analogous result (discrete counter part) by Candes and Fernandez-Granda in [54] shows that TV minimization succeeds as long as $OSR > 4$. So the above theorem improves on this result by a factor of roughly 1.25. In addition to being sharper our proofs are also significantly shorter. Furthermore, since our proofs are based on the construction of approximate dual certificates we believe that our proof will allow rigorous proofs of many other results such as stability to sparse corruption. In the author's view this is the "Golfing scheme" way of building dual certificates for highly coherent dictionaries. It is well known that the Golfing scheme not only simplified the proofs for matrix completion it also enabled many other results such as the proof of RPCA [56]. The author believes that the approximate dual construction here can also be utilized in a similar manner.

Finally, we would like to mention the more precise result below.

Theorem 18.2.2 *Let $\Omega \geq 10$ (18.1.1) yields exact recovery as long as*

$$OSR \geq \eta(\Omega),$$

where $\eta(\Omega)$ is depicted in Figure 18.1.

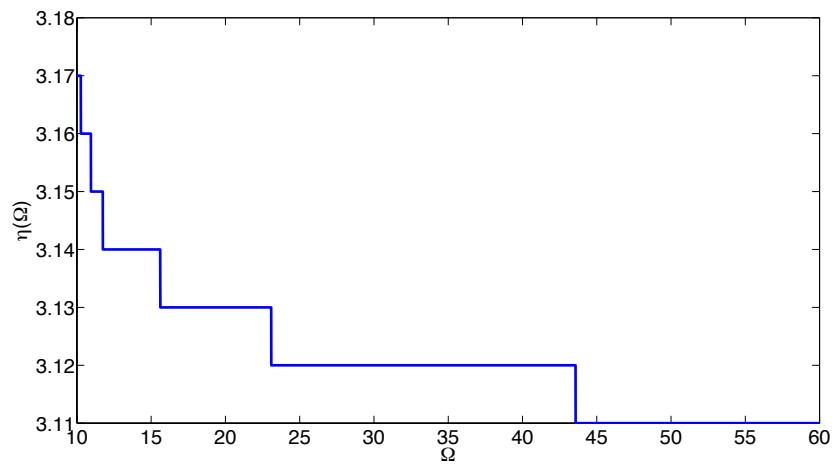


Figure 18.1: Plot of minimum possible OSR ($\eta(\Omega)$) under which TV minimization yields exact recovery as a function of the highest frequency Ω .

Chapter 19

Proofs

19.1 Proof of compressive sensing with coherent dictionaries (Theorem 17.2.4)

To prove our result we shall first state a lemma from the geometric functional analysis literature.

Lemma 19.1.1 (Low M^* estimate) *Let \mathcal{D} be a symmetric convex body in \mathbb{R}^n , and let $E \in G_{n,n-k}$ be a random subspace of codimension k . Then, with probability at least $1 - e^{-k}$,*

$$\text{diam}(\mathcal{D} \cap E) \leq c \sqrt{\frac{n}{k}} M^*(\mathcal{D}), \quad (19.1.1)$$

where c is a global constant.

We establish our result by a geometric null space property stated in the lemma below and proven later on in this section.

Lemma 19.1.2 (Geometric Null Space Property)

Defining $\mathcal{F} = \text{Face}(\mathbf{f})$, assume that¹

$$\text{cone}(\mathcal{F}) \cap \text{Null}(\mathbf{A}) = \{\mathbf{0}\}. \quad (19.1.2)$$

If

$$\text{dist}\left(\text{aff}(\mathcal{P}_{\mathbf{A}}\mathcal{F}), \text{symconv}(\mathcal{P}_{\mathbf{A}}[\mathbf{D}_{s+1}, \dots, \mathbf{D}_N])\right) > 0 \quad (19.1.3)$$

then ℓ_1 synthesis procedure defined in (17.1.1) can recover every \mathbf{f} that belongs to \mathcal{F} exactly.

The reverse of this lemma is true as well. That is, the condition is both necessary and sufficient. However, the argument is a bit more involved and therefore we skip the details.

We note that as stated in the footnote the first condition is a very mild condition and is only to ensure that the face \mathcal{F} does not “fold on itself”, which is necessary for unique recovery of any signal $\mathbf{f} \in \mathcal{F}$. The second condition says that the distance of the face \mathcal{F} with respect to the polytope \mathcal{D} after projection onto the row space of \mathbf{A} must be positive, further highlighting the importance of this distance definition in compressed sensing over general polytopes. A more intuitive argument of this fact is provided with a pictorial depiction in Section 19.1.2.

With all these auxiliary lemmas in place we turn our attention to the proof of our main result. The proof is by contradiction, assume that \mathbf{f} can not be recovered using the ℓ_1 synthesis procedure of (17.1.1). Then according to the geometric null space property we proved earlier

$$\text{dist}\left(\text{aff}(\mathcal{P}_{\mathbf{A}}\mathcal{F}), \text{symconv}(\mathcal{P}_{\mathbf{A}}[\mathbf{D}_{s+1}, \dots, \mathbf{D}_N])\right) = 0. \quad (19.1.4)$$

therefore, there must be a vector $\mathbf{f}_1 \in \mathcal{F}$ and a vector $\mathbf{f}_2 \in \text{conv}(\pm \mathbf{D}_{s+1}, \dots, \pm \mathbf{D}_N)$ such that $\mathbf{h} = \mathbf{f}_2 - \mathbf{f}_1 \in \text{Null}(\mathbf{A})$. Now notice that the vector \mathbf{h} has both end points in the

¹Notice that this condition is easily satisfied for faces of low dimension e.g. one could also consider the stronger assumption of $\text{span}(\mathbf{D}_T) \cap \text{Null}(\mathbf{A}) = \{\mathbf{0}\}$, which is easily satisfied for i.i.d. gaussian, i.i.d. ± 1 etc.

polytope \mathcal{D} . Hence,

$$\|\mathbf{h}\|_{\ell_2} \leq \text{diam}(2\mathcal{D} \cap \text{Null}(\mathbf{A})) \quad (19.1.5)$$

Now using the Low M^* estimate Lemma stated above we get that with probability at least $1 - e^{-m}$

$$\|\mathbf{h}\|_{\ell_2} \leq c\sqrt{\frac{n}{m}}M^*(\mathcal{D}) \quad (19.1.6)$$

Now notice that

$$\|\mathbf{h}\|_{\ell_2} = \|\mathbf{f}_2 - \mathbf{f}_1\|_{\ell_2} \geq \text{dist}(\mathcal{F}, \mathcal{D}) \quad (19.1.7)$$

therefore we get that

$$m \leq cn \left(\frac{M^*(\mathcal{D})}{\text{dist}(\mathcal{F}, \mathcal{D})} \right)^2 \quad (19.1.8)$$

which is a contradiction, concluding the proof.

19.1.1 Proof of the geometric null space property (Lemma 19.1.2)

To prove this lemma we assume that the distance is positive and we want to prove that the optimization problem yields the exact solution \mathbf{f} . Notice that proving this is exactly equivalent to providing a dual certificate for this problem. However the general form of this dual certificate is given by:

$$\boldsymbol{\nu} \in \text{row}(\mathbf{A}) \quad (19.1.9)$$

$$\tilde{\mathbf{D}}_T^* \boldsymbol{\nu} = 1, \quad (19.1.10)$$

$$\|\mathbf{D}_{T^c}^* \boldsymbol{\nu}\|_{\ell_\infty} < 1. \quad (19.1.11)$$

where $\tilde{\mathbf{D}}$ and $\mathbf{1}$ is the all one vector of dimension s . Now the important question is what direction should one choose for this dual certificate? We will pick the direction $\hat{\boldsymbol{\nu}}$ along which the minimal distance of (19.1.3) is obtained. More precisely, $\boldsymbol{\nu}$ is in the direction of $\boldsymbol{\nu}_2 - \boldsymbol{\nu}_1$ with $\boldsymbol{\nu}_1 \in \text{aff}(\mathcal{P}_A \mathcal{F})$ and $\boldsymbol{\nu}_2 \in \text{symconv}(\mathcal{P}_A [\mathbf{D}_{s+1}, \dots, \mathbf{D}_N])$ obeying

$$\|\boldsymbol{\nu}_2 - \boldsymbol{\nu}_1\|_{\ell_2} = \text{dist}\left(\text{aff}(\mathcal{P}_A \mathcal{F}), \text{symconv}(\mathcal{P}_A [\mathbf{D}_{s+1}, \dots, \mathbf{D}_N])\right).$$

By definition this direction belongs to the row space of the matrix \mathbf{A} , therefore any vector along this direction satisfies the first of the three dual certificate conditions. With the direction of the dual certificate fixed the question is how should one choose the length of this dual certificate, the second dual certificate condition guides us towards this value, i.e. choose the dual certificate $\boldsymbol{\nu}$ in the direction of $\boldsymbol{\nu}_2 - \boldsymbol{\nu}_1$ and such that

$$\tilde{\mathbf{D}}_1^* \boldsymbol{\nu} = 1. \quad (19.1.12)$$

where we use $\tilde{\mathbf{D}}_1$ to refer to the first column of $\tilde{\mathbf{D}}_T$.

With the dual certificate fixed now we must check that the second and third conditions for the dual certificate are satisfied. For this purpose, notice that the direction of minimum distance (direction of $\boldsymbol{\nu}$) is orthogonal to $\text{aff}(\mathcal{P}_A(\mathcal{F})) = \text{aff}(\mathcal{P}_A(\tilde{\mathbf{D}}_T))$. This implies that for any $j \neq k$ in $\{1, \dots, s\}$ we have

$$\begin{aligned} \boldsymbol{\nu} &\perp \mathcal{P}_A(\tilde{\mathbf{D}}_j - \tilde{\mathbf{D}}_k) \Rightarrow \langle \boldsymbol{\nu}, \tilde{\mathbf{D}}_j - \tilde{\mathbf{D}}_k \rangle \stackrel{\mathcal{P}_A(\boldsymbol{\nu})=\boldsymbol{\nu}}{=} \langle \boldsymbol{\nu}, \mathcal{P}_A(\tilde{\mathbf{D}}_j - \tilde{\mathbf{D}}_k) \rangle = 0, \\ &\Rightarrow \tilde{\mathbf{D}}_s^* \boldsymbol{\nu} = \dots = \tilde{\mathbf{D}}_1^* \boldsymbol{\nu} = 1, \\ &\Rightarrow \tilde{\mathbf{D}}_T^* \boldsymbol{\nu} = \mathbf{1}, \end{aligned}$$

and therefore the second dual certificate condition is also satisfied. Now to check the third dual certificate condition notice that for every $\mathbf{f}_1 \in \text{aff}(\mathcal{F})$ with similar

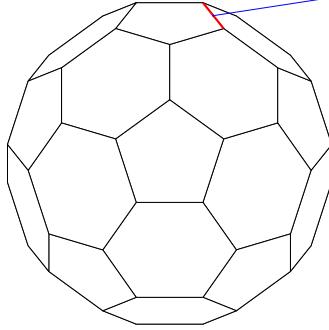


Figure 19.1: Edge in red denotes face of polytope. Line in blue denotes null space of \mathbf{A} . Null space intersects with the interior of the polytope (synthesis fails).

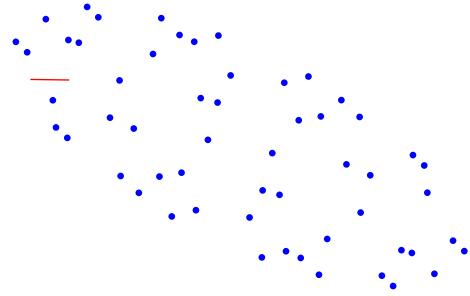


Figure 19.2: 2-D plane representing the row space of \mathbf{A} , projection of the face in red onto the row space of \mathbf{A} , and projection of the rest of the vertices of the polytope onto this subspace in blue.

arguments to the above we have $\langle \boldsymbol{\nu}, \mathbf{f}_1 \rangle = 1$, therefore

$$\begin{aligned}
1 - \|\mathbf{D}_{T^c}^* \boldsymbol{\nu}\|_{\ell_\infty} &= \langle \boldsymbol{\nu}, \mathbf{f}_1 \rangle - \max_{j \in T^c} \max_{\phi_j \in [0, 2\pi)} \langle \boldsymbol{\nu}, e^{i\phi_j} \mathbf{D}_j \rangle \\
&\stackrel{\mathcal{P}_{\mathbf{A}}(\boldsymbol{\nu})=\boldsymbol{\nu}}{=} \underbrace{\min_{\mathbf{f}_1 \in \text{aff}(\mathcal{P}_{\mathbf{A}}\mathcal{F})} \langle \boldsymbol{\nu}, \mathbf{f}_1 \rangle - \max_{\mathbf{f}_2 \in \text{symconv}(\mathcal{P}_{\mathbf{A}}[\mathbf{D}_{s+1}, \dots, \mathbf{D}_N])} \langle \boldsymbol{\nu}, \mathbf{f}_2 \rangle}_{\langle \boldsymbol{\nu}, \mathbf{f}_1 - \mathbf{f}_2 \rangle} \\
&= \min_{\mathbf{f}_1 \in \text{aff}(\mathcal{P}_{\mathbf{A}}\mathcal{F}), \mathbf{f}_2 \in \text{symconv}(\mathcal{P}_{\mathbf{A}}[\mathbf{D}_{s+1}, \dots, \mathbf{D}_N])} \langle \boldsymbol{\nu}, \mathbf{f}_1 - \mathbf{f}_2 \rangle \\
&= \|\boldsymbol{\nu}\|_{\ell_2} \text{dist}\left(\text{aff}(\mathcal{P}_{\mathbf{A}}\mathcal{F}), \text{symconv}(\mathcal{P}_{\mathbf{A}}[\mathbf{D}_{s+1}, \dots, \mathbf{D}_N])\right) > 0
\end{aligned}$$

Therefore $\boldsymbol{\nu}$ is a legitimate dual certificate, concluding the proof of the forward direction.

19.1.2 Interpretation of the geometric null space property

Now that the statement and proof of the geometric null space property is complete we shall try to interpret it. For this purpose consider Figs. 19.1 and 19.2. These figures denote the problem when the polytope \mathcal{D} is given by a bucky-ball (depicted in Fig. 19.1). In this case the assumption is that the original signal $\mathbf{f} \in \mathbb{R}^3$ and $\frac{\mathbf{f}}{\|\mathbf{f}\|_D}$ lives on a

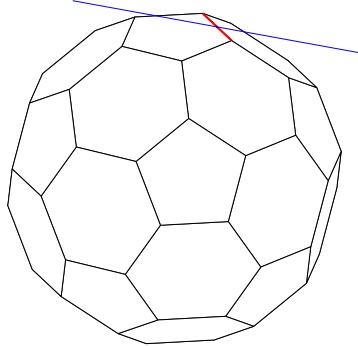


Figure 19.3: Edge in red denotes face of polytope. Line in blue denotes null space of \mathbf{A} . Null space is tangent to the space of \mathbf{A} . Null space is tangent to the polytope (synthesis succeeds).



Figure 19.4: 2-D plane representing the row space of \mathbf{A} , projection of the face in red onto the row space of \mathbf{A} , and projection of the rest of the vertices of the polytope onto this subspace in blue.

2-dimensional face of the polytope \mathcal{D} . The sensing matrix $\mathbf{A} \in \mathbb{R}^{2 \times 3}$ is i.i.d. gaussian. Fig. 19.1 depicts the polytope along with the signal \mathbf{f} and the corresponding face (shown in red). The null space of the matrix \mathbf{A} is shown here in blue. As can be seen the null space intersects with the interior of the polytope. Therefore in this case the synthesis recovery procedure will not be successful in recovering the signal. Fig. 19.4 shows the projection of this polytope on the row space of the matrix \mathbf{A} . In this figure, the projection of the face is shown by a red segment and the projection of the rest of the vertices of the polytope are shown in blue. As can be seen the affine hull of the projection of the red segment (imagine a red line passing through that segment), intersects with the convex hull of the blue points; i.e. the distance between these two sets is zero. This matches the above stated Geometric Null space property, confirming that recovery is not possible in this case.

Figs. 19.3 and 19.4 shows the same recovery problem using another instance of the random matrix \mathbf{A} . As can be seen in 19.3 in this case the null space of \mathbf{A} is tangent to the polytope, and therefore the synthesis approach will be successful. Fig. 19.4 shows that the affine subspace passing through the projection of the face (line of segment in red), does not intersect with the convex hull of the rest of the points, again confirming the geometric null space property stated above.

19.2 Proof of Lemma 17.2.5

We will use F to denote the translation of the affine subspace to the origin (subspace with same orientation as $\text{aff}(\mathcal{F})$ but passing through the origin). We will use F^\perp to denote the orthogonal complement of F . Notice that $\tilde{\mathbf{D}}_T^* \boldsymbol{\nu} = \mathbf{1}$ implies $\boldsymbol{\nu} \in F^\perp$. Defining $\mathbf{u} = \frac{\boldsymbol{\nu}}{\|\boldsymbol{\nu}\|_{\ell_2}}$ we have

$$\max_{\boldsymbol{\nu} \in \mathbb{R}^n : \tilde{\mathbf{D}}_T^* \boldsymbol{\nu} = \mathbf{1}} \frac{1 - \|\mathbf{D}_{T^c}^* \boldsymbol{\nu}\|_{\ell_\infty}}{\|\boldsymbol{\nu}\|_{\ell_2}} = \max_{\boldsymbol{\nu} \in \mathbb{R}^n : \boldsymbol{\nu} \in F^\perp} \frac{\tilde{\mathbf{D}}_1^* \boldsymbol{\nu} - \|\mathbf{D}_{T^c}^* \boldsymbol{\nu}\|_{\ell_\infty}}{\|\boldsymbol{\nu}\|_{\ell_2}} = \max_{\mathbf{u} \in \mathbb{R}^n : \mathbf{u} \in F^\perp, \|\mathbf{u}\|_{\ell_2} = 1} \tilde{\mathbf{D}}_1^* \mathbf{u} - \|\mathbf{D}_{T^c}^* \mathbf{u}\|_{\ell_\infty}$$

It is easy to see that relaxing the constraint $\|\mathbf{u}\|_{\ell_2} = 1$ to $\|\mathbf{u}\|_{\ell_2} \leq 1$ does not change anything, putting the optimization problem into epigraph form we arrive at the equivalent problem

$$\begin{aligned} & \underset{t, \mathbf{u}}{\text{maximize}} \quad t \\ & \text{subject to} \quad \forall j \in T^c : (\tilde{\mathbf{D}}_1 - \mathbf{D}_j)^* \mathbf{u} \leq t \\ & \quad \forall j \in T^c : (\tilde{\mathbf{D}}_1 + \mathbf{D}_j)^* \mathbf{u} \leq t \\ & \quad \mathbf{u} \in F^\perp, \|\mathbf{u}\|_{\ell_2} \leq 1. \end{aligned}$$

We will use $\boldsymbol{\mu}, \boldsymbol{\lambda} \in \mathbb{R}^{N-s}$ with $\boldsymbol{\mu}, \boldsymbol{\lambda} \geq 0$, $\boldsymbol{\alpha} \in F$ and $\beta \in \mathbb{R}$ (or the complex versions). We now try to derive the dual of the above problem, to this aim we have the unconstrained version

$$\begin{aligned} & \underset{t, \mathbf{u}}{\text{maximize}} \quad t + \sum_{j \in T^c} \boldsymbol{\mu}_j [(\tilde{\mathbf{D}}_1 - \mathbf{D}_j)^* \mathbf{u} - t] + \sum_{j \in T^c} \boldsymbol{\lambda}_j [(\tilde{\mathbf{D}}_1 + \mathbf{D}_j)^* \mathbf{u} - t] + \langle \boldsymbol{\alpha}, \mathbf{u} \rangle + \beta (\|\mathbf{u}\|_{\ell_2} - 1) \\ & \text{subject to} \quad \boldsymbol{\mu}, \boldsymbol{\lambda} \geq 0, \quad \boldsymbol{\alpha} \in F. \end{aligned}$$

Maximizing the expression with respect to t leads to the conclusion that $\mathbf{1}^* \boldsymbol{\mu} + \mathbf{1}^* \boldsymbol{\lambda} = 1$ and the expression above simplifies to

$$\begin{aligned} & \underset{\mathbf{u}}{\text{maximize}} \quad [\boldsymbol{\alpha} + \tilde{\mathbf{D}}_1 - \mathbf{D}_{T^c}(\boldsymbol{\mu} - \boldsymbol{\lambda})]^* \mathbf{u} + \beta (\|\mathbf{u}\|_{\ell_2} - 1) \\ & \text{subject to} \quad \boldsymbol{\mu}, \boldsymbol{\lambda} \geq 0, \quad \mathbf{1}^* \boldsymbol{\mu} + \mathbf{1}^* \boldsymbol{\lambda} = 1, \quad \boldsymbol{\alpha} \in \mathcal{F}. \end{aligned}$$

Now notice that $\mathbf{v} = \boldsymbol{\alpha} + \tilde{\mathbf{D}}_1$ with $\boldsymbol{\alpha} \in F$ denotes any point in $\text{aff}(\mathcal{F})$ and noticing that the objective will be $-\infty$ if $\|\boldsymbol{\alpha} + \tilde{\mathbf{D}}_1 - \mathbf{D}_{T^c}(\boldsymbol{\mu} - \boldsymbol{\lambda})\|_{\ell_2} > \beta$ therefore we arrive at the following dual problem

$$\begin{aligned} & \underset{\boldsymbol{\mu}, \boldsymbol{\lambda}, \mathbf{v}}{\text{minimize}} \quad \|\mathbf{v} - \mathbf{D}_{T^c}(\boldsymbol{\mu} - \boldsymbol{\lambda})\|_{\ell_2} \\ & \text{subject to} \quad \boldsymbol{\mu}, \boldsymbol{\lambda} \geq 0, \quad \mathbf{1}^* \boldsymbol{\mu} + \mathbf{1}^* \boldsymbol{\lambda} = 1, \quad \mathbf{v} \in \text{aff}(\mathcal{F}) \end{aligned}$$

Which can equivalently be written in the form

$$\begin{aligned} & \underset{\boldsymbol{\theta}, \mathbf{v}}{\text{minimize}} \quad \|\mathbf{v} - \mathbf{D}_{T^c} \boldsymbol{\theta}\|_{\ell_2} \\ & \text{subject to} \quad \|\boldsymbol{\theta}\|_{\ell_1} \leq 1, \quad \mathbf{v} \in \text{aff}(\mathcal{F}) \end{aligned}$$

The latter is exactly the definition of $\text{dist}(\mathcal{F}, \mathcal{D})$, completing the proof.

19.3 Proof of denoising with coherent dictionaries (Theorem 17.2.6)

Let $\boldsymbol{\nu}$ be a dual certificate of the face \mathcal{F} to prove Theorem 17.2.6 we shall establish the stronger upper bound below.

$$\|\hat{\mathbf{f}} - \mathbf{f}_T\|_{\ell_2}^2 \leq 34 \|\mathbf{f}_{T^c}\|_{\ell_2}^2 + 32 \frac{s}{n} \sigma^2 + 128C^2 \frac{\sigma^2}{n} \frac{\log N}{\left(\frac{1 - \|\mathbf{D}_{T^c}^* \boldsymbol{\nu}\|_{\ell_\infty}}{\|\boldsymbol{\nu}\|_{\ell_2}} \right)^2}, \quad (19.3.1)$$

The result of the theorem follows by choosing the dual certificate that maximizes the ratio $\frac{1 - \|\mathbf{D}_{T^c}^* \boldsymbol{\nu}\|_{\ell_\infty}}{\|\boldsymbol{\nu}\|_{\ell_2}}$. We thus turn our attention to proving (19.3.1).

Defining $\mathbf{h} = \hat{\mathbf{x}} - \mathbf{x}_T$ standard calculations yield

$$\begin{aligned} \frac{1}{2} \|\mathbf{D}\mathbf{h}\|_{\ell_2}^2 + \lambda \|\mathbf{h}_{T^c}\|_{\ell_1} & \leq \langle \mathbf{D}\mathbf{h}, \mathbf{w} + \mathbf{f}_{T^c} \rangle - \lambda \langle \mathbf{h}_T, \mathbf{D}_T^* \boldsymbol{\nu} \rangle \\ & = \langle \mathbf{D}\mathbf{h}, \mathbf{w} + \mathbf{f}_{T^c} \rangle - \lambda \langle \mathbf{D}\mathbf{h}, \boldsymbol{\nu} \rangle + \lambda \langle \mathbf{D}_{T^c} \mathbf{h}_{T^c}, \boldsymbol{\nu} \rangle \quad (19.3.2) \end{aligned}$$

using $ab < \frac{1}{2}a^2 + \frac{1}{2}b^2$ with $a = \frac{\|\mathbf{D}\mathbf{h}\|_{\ell_2}}{2\sqrt{\lambda}}$ and $b = 2\sqrt{\lambda}\|\boldsymbol{\nu}\|_{\ell_2}$ we conclude that

$$-\lambda\langle \mathbf{D}\mathbf{h}, \boldsymbol{\nu} \rangle \leq \frac{1}{8}\|\mathbf{D}\mathbf{h}\|_{\ell_2}^2 + 2\lambda\|\boldsymbol{\nu}\|_{\ell_2}^2.$$

Also, by Holder's inequality we have $\lambda\langle \mathbf{D}_{T^c}\mathbf{h}_{T^c}, \boldsymbol{\nu} \rangle \leq \alpha\lambda\|\mathbf{h}_{T^c}\|_{\ell_1}$. Plugging the latter two inequalities in (19.3.2) we arrive at

$$\frac{3}{8}\|\mathbf{D}\mathbf{h}\|_{\ell_2}^2 + (1 - \alpha)\lambda\|\mathbf{h}_{T^c}\|_{\ell_1} \leq \langle \mathbf{D}\mathbf{h}, \mathbf{w} + \mathbf{f}_{T^c} \rangle + 2\lambda^2\|\boldsymbol{\nu}\|_{\ell_2}^2$$

Define \mathbf{P} as the projection onto $\text{span}(\tilde{\mathbf{D}}_T)$. We have

$$\langle \mathbf{D}\mathbf{h}, \mathbf{w} \rangle = \langle \mathbf{D}\mathbf{h}, \mathbf{P}\mathbf{w} \rangle + \langle \mathbf{D}\mathbf{h}, (\mathbf{I} - \mathbf{P})\mathbf{w} \rangle$$

We now bound each of these terms

$$\langle \mathbf{D}\mathbf{h}, \mathbf{P}\mathbf{w} \rangle \leq \frac{1}{8}\|\mathbf{D}\mathbf{h}\|_{\ell_2}^2 + 2\|\mathbf{P}\mathbf{w}\|_{\ell_2}^2 \leq \frac{1}{8}\|\mathbf{D}\mathbf{h}\|_{\ell_2}^2 + 2\frac{s}{n}\sigma^2$$

Similarly, it is a standard probability calculation to show that with high probability

$$\langle \mathbf{D}\mathbf{h}, (\mathbf{I} - \mathbf{P})\mathbf{w} \rangle = \langle \mathbf{D}_{T^c}\mathbf{h}_{T^c}, (\mathbf{I} - \mathbf{P})\mathbf{w} \rangle = \langle \mathbf{h}_{T^c}, \mathbf{D}_{T^c}^*(\mathbf{I} - \mathbf{P})\mathbf{w} \rangle \leq 2\sqrt{\frac{\log N}{n}}\sigma\|\mathbf{h}_{T^c}\|_{\ell_1}$$

Also, we have

$$\langle \mathbf{D}\mathbf{h}, \mathbf{f}_{T^c} \rangle \leq \frac{1}{8}\|\mathbf{D}\mathbf{h}\|_{\ell_2}^2 + 2\|\mathbf{f}_{T^c}\|_{\ell_2}^2.$$

Putting all this together we arrive at

$$\frac{1}{8}\|\mathbf{D}\mathbf{h}\|_{\ell_2}^2 + \left[(1 - \alpha)\lambda - 2\sqrt{\frac{\log N}{n}}\sigma\right]\|\mathbf{h}_{T^c}\|_{\ell_1} \leq 2\|\mathbf{f}_{T^c}\|_{\ell_2}^2 + 2\frac{s}{n}\sigma^2 + 2\lambda^2\|\boldsymbol{\nu}\|_{\ell_2}^2$$

Using the specific choice of λ we have

$$\|\hat{\mathbf{f}} - \mathbf{f}_T\|_{\ell_2}^2 \leq 16\|\mathbf{f}_{T^c}\|_{\ell_2}^2 + 16\frac{s}{n}\sigma^2 + 64C^2\frac{\sigma^2}{n} \frac{\log N}{\left(\frac{1 - \|\mathbf{D}_{T^c}^*\boldsymbol{\nu}\|_{\ell_\infty}}{\|\boldsymbol{\nu}\|_{\ell_2}}\right)^2}.$$

Using the inequality $\|\mathbf{a} + \mathbf{b}\|_{\ell_2}^2 \leq 2\|\mathbf{a}\|_{\ell_2}^2 + 2\|\mathbf{b}\|_{\ell_2}^2$, with $\mathbf{a} = \hat{\mathbf{f}} - \mathbf{f}_T$ and $\mathbf{b} = -\mathbf{f}_{T^c}$ we arrive at the result.

19.4 Proof of sparse recovery with highly coherent dictionaries

19.4.1 Proof of the connection (Theorem 18.1.5)

Assume that $x_d(t) = \sum_{s \in S} x[s] \delta(t - s)$ is a signal in the interval $[0,1]$ of the form of the discrete frequency model satisfying $\Delta_d(S) \geq \eta$. We wish to show that the solution to the optimization problem (18.1.2) is unique and is equal to $x_d(t)$. We prove this by contradiction assume there is another signal $\bar{x}_d(t)$ such that the first $(2f_c + 1)$ coefficients of its Fourier series agree with that of $x_d(t)$ but has smaller total variation norm. That is

$$\|\bar{x}_d\|_{TV} \leq \|x_d\|_{TV} \quad \text{and} \quad \bar{X}(f) = X(f) \quad \text{for } f = 0, \pm 1, \pm 2, \dots \pm f_c.$$

First define $y[s] = \frac{x[s]}{(\sum_{n=-\infty}^{n=+\infty} |g(s-n)|)}$ (we use a g that is summable like $g(t) = \text{sinc}^2(2t) + \text{sinc}^2(2t - 1/2)$) and set

$$\begin{aligned} y_d(t) &= \sum_{s \in S} y_d[s] \delta(t - s) := \sum_{s \in S} \frac{x_d[s]}{(\sum_{n=-\infty}^{n=+\infty} |g(s-n)|)} \delta(t - s), \\ \bar{y}_d(t) &= \sum_{s \in S} \bar{y}_d[s] \delta(t - s) := \sum_{s \in S} \frac{\bar{x}_d[s]}{(\sum_{n=-\infty}^{n=+\infty} |g(s-n)|)} \delta(t - s). \end{aligned}$$

Now define the following two continuous signals corresponding to $x_d(t)$ and $\bar{x}_d(t)$

$$\begin{aligned} x_c(t) &= g(t) \sum_{n=-\infty}^{\infty} y_d(t - n), \\ \bar{x}_c(t) &= g(t) \sum_{n=-\infty}^{\infty} \bar{y}_d(t - n). \end{aligned}$$

Set $\Omega = f_c$ and notice that by construction $\bar{X}(f) = X(f)$ for $f \in [-\Omega, \Omega]$ and also that $OSR_c(S) = OSR_d(S) \geq \eta$. However, we have

$$\|\bar{x}_c\|_{TV} = \|\bar{x}_d\|_{TV} \leq \|x_d\|_{TV} = \|x_c\|_{TV}$$

However, the latter is in contradiction with our assumption that (18.1.1) yields exact recovery for any signal obeying $OSR_c(S) \geq \eta$.

19.4.2 Proof of the continuous super-resolution problem (Theorems 18.2.1 and 18.2.2)

The proof is stated for a slightly weaker result. The above result is obtained by improving Lemma 19.4.4 by actually drawing the bound of that lemma in the computer. Let h be a high-pass measure whose Fourier transform (denoted by $H(f)$) vanishes in an interval $[-\Omega, \Omega]$. Define the following kernels

$$k(t) = \frac{\cos(2\pi\Omega t)}{1 - 16\Omega^2 t^2} \Leftrightarrow K(f) = \begin{cases} \frac{\pi}{4\Omega} \cos \frac{\pi}{2\Omega} f & |f| \leq \Omega \\ 0 & |f| < \Omega \end{cases}.$$

$$g(t) = \frac{1}{2\pi\Omega} \frac{\sin(2\pi\Omega t)}{1 - 4\Omega^2 t^2} \Leftrightarrow G(f) = \begin{cases} \frac{1}{4\Omega^2} \sin \frac{\pi}{\Omega} f & |f| \leq \Omega \\ 0 & |f| < \Omega \end{cases}.$$

Note that we have

$$\begin{aligned} k'(t) &= 2\Omega \frac{\pi \sin(2\pi\Omega t)(16\Omega^2 t^2 - 1) + 16\Omega t \cos(2\pi\Omega t)}{(1 - 16\Omega^2 t^2)^2} \\ k''(t) &= 4\Omega^2 \frac{32\pi\Omega t(1 - 16t^2\Omega^2) \sin(2\pi\Omega t) + (\pi^2(1 - 16\Omega^2 t^2)^2 - 8(48\Omega^2 t^2 + 1)) \cos(2\pi\Omega t)}{(16\Omega^2 t^2 - 1)^3}. \end{aligned}$$

$$\begin{aligned} g'(t) &= \frac{\pi(4\Omega^2 t^2 - 1) \cos(2\pi\Omega t) - 4\Omega t \sin(2\pi\Omega t)}{\pi(4\Omega^2 t^2 - 1)^2} \\ g''(t) &= 2\Omega \frac{(\pi^2(4\Omega^2 t^2 - 1)^2 - 24\Omega^2 t^2 - 2) \sin(2\pi\Omega t) + 4\pi\Omega t(4\Omega^2 t^2 - 1) \cos(2\pi\Omega t)}{\pi(4\Omega^2 t^2 - 1)^3}. \end{aligned}$$

We use $\lambda_c = 1/(2\Omega)$ and define the following normalized versions of the kernels and its derivatives.

$$\begin{aligned} k_n(\gamma) &= \frac{\cos(\pi\gamma)}{1 - 4\gamma^2} \\ k'_n(\gamma) &= \frac{\pi \sin(\pi\gamma)(4\gamma^2 - 1) + 8\gamma \cos(\pi\gamma)}{(1 - 4\gamma^2)^2} \\ k''_n(\gamma) &= \frac{16\pi\gamma(1 - 4\gamma^2) \sin(\pi\gamma) + (\pi^2(1 - 4\gamma^2)^2 - 8(12\gamma^2 + 1)) \cos(\pi\gamma)}{(4\gamma^2 - 1)^3}. \end{aligned}$$

$$\begin{aligned} g_n(\gamma) &= \frac{\lambda_c \sin(\pi\gamma)}{\pi(1 - \gamma^2)} \\ g'_n(\gamma) &= \lambda_c \frac{\pi(\gamma^2 - 1) \cos(\pi\gamma) - 2\gamma \sin(\pi\gamma)}{\pi(\gamma^2 - 1)^2} \\ g''_n(\gamma) &= \lambda_c \frac{(\pi^2(\gamma^2 - 1)^2 - 6\gamma^2 - 2) \sin(\pi\gamma) + 4\pi\gamma(\gamma^2 - 1) \cos(\pi\gamma)}{\pi(\gamma^2 - 1)^3}. \end{aligned}$$

Using the change of variable $\gamma = 2\Omega t$ note that we have

$$k(t) = k_n(\gamma), \quad k'(t) = 2\Omega k'_n(\gamma), \quad k''(t) = 4\Omega^2 k''_n(\gamma).$$

$$g(t) = g_n(\gamma), \quad g'(t) = 2\Omega g'_n(\gamma), \quad g''(t) = 4\Omega^2 g''_n(\gamma).$$

We partition the real line as follows.

Definition 19.4.1 For $s \in S$ define

$$\mathcal{N}(s) = \{t : |t - s| \leq \gamma_c \lambda_c\}.$$

Definition 19.4.2

$$\begin{aligned}\mathcal{N} &= \cup_{s \in S} N(s), \\ \mathcal{F} &= \mathbb{R} \setminus \mathcal{N}.\end{aligned}$$

Set $\epsilon(s) = |h(s)| / h(s)$ and define

$$\begin{aligned}v(t) &= \sum_{s' \in S} \epsilon(s') k(t - s'), \\ &= \epsilon(s) k(t - s) + \sum_{s' \in S \setminus s} \epsilon(s') k(t - s'), \\ &:= \epsilon(s) k(t - s) + u_s(t).\end{aligned}$$

We have

$$\int K(f) e^{-2\pi i fs} H(df) = \int k(t - s) h(dt).$$

Using this equality together with the fact that h is high pass we have

$$0 = \int K(f) e^{-2\pi i fs} H(df) = h(s) + \int_{t \neq s} k(t - s) h(dt).$$

Note that $h(s)$ is the delta part of h at s . We multiply each of these relationships by the sign $\epsilon(s) = |h(s)| / h(s)$ and sum over entries in S . We thus arrive at

$$0 = \sum_{s \in S} |h(s)| + \sum_{s \in S} \int_{t \neq s} \epsilon(s) k(t - s) h(dt).$$

We thus have

$$\begin{aligned}
\sum_{s \in S} |h(s)| &= - \sum_{s \in S} \int_{t \neq s} \epsilon(s) k(t-s) h(dt), \\
&= - \sum_{s \in S} \left(\int_{N(s) \setminus \{s\}} \epsilon(s) k(t-s) h(dt) \right. \\
&\quad \left. + \sum_{s' \in S \setminus \{s\}} \int_{N(s')} \epsilon(s) k(t-s) h(dt) + \int_{\mathcal{F}} \epsilon(s) k(t-s) h(dt) \right), \\
&= - \sum_{s \in S} \int_{N(s) \setminus \{s\}} \epsilon(s) k(t-s) h(dt) - \sum_{s' \in S} \sum_{s \in S \setminus \{s'\}} \int_{N(s')} \epsilon(s) k(t-s) h(dt) \\
&\quad - \int_{\mathcal{F}} v(t) h(dt), \\
&= - \sum_{s \in S} \int_{N(s) \setminus \{s\}} \epsilon(s) k(t-s) h(dt) - \sum_{s' \in S} \int_{N(s')} \left(\sum_{s \in S \setminus \{s'\}} \epsilon(s) k(t-s) \right) h(dt) \\
&\quad - \int_{\mathcal{F}} v(t) h(dt), \\
&= - \sum_{s \in S} \int_{N(s) \setminus \{s\}} \epsilon(s) k(t-s) h(dt) - \sum_{s' \in S} \int_{N(s')} u_{s'}(t) h(dt) - \int_{\mathcal{F}} v(t) h(dt), \\
&= - \sum_{s \in S} \int_{N(s) \setminus \{s\}} \epsilon(s) k(t-s) h(dt) - \sum_{s \in S} \int_{N(s)} u_s(t) h(dt) - \int_{\mathcal{F}} v(t) h(dt).
\end{aligned} \tag{19.4.1}$$

Definition 19.4.3 We define the following moments

$$\begin{aligned}
\mathcal{I}_{N(s)}^{(0)} &= \left| \int_{t \in N(s)} h(dt) \right|, \\
\mathcal{I}_{N(s)}^{(1)} &= \frac{1}{\lambda_c} \left| \int_{t \in N(s)} (t-s) h(dt) \right|, \\
\mathcal{I}_{N(s)}^{(2)} &= \frac{1}{2\lambda_c^2} \int_{t \in N(s)} (t-s)^2 |h|(dt).
\end{aligned}$$

$$\mathcal{I}_N^{(0)} = \sum_{s \in S} \mathcal{I}_{N(s)}^{(0)}, \quad \mathcal{I}_N^{(1)} = \sum_{s \in S} \mathcal{I}_{N(s)}^{(1)}, \quad \mathcal{I}_N^{(2)} = \sum_{s \in S} \mathcal{I}_{N(s)}^{(2)}.$$

We have the following result

Lemma 19.4.4 If $OSR \geq 1$,

$$\max_{t \in \mathcal{F}} |v(t)| \leq 1 - \beta,$$

where

$$\beta = 1 - k_u^{(0)}(\gamma_c) - k_u^{(0)}(OSR/2) - f_0(2OSR, -OSR) - f_0(2OSR, -2\gamma_c).$$

Here, $k_u^{(0)}$ and f_0 are real valued functions whose exact form shall be defined in the proof.

Lemma 19.4.5 *For $t \in \mathcal{N}(0)$ and $\gamma_c \leq 0.81$ we have*

$$|k(t)| \leq 1 - \frac{1}{2\lambda_c^2} |k_n''(\gamma_c)| t^2.$$

We continue the chain of inequalities (19.4.1) and apply

$$\begin{aligned} \|\mathcal{P}_S(\mathbf{h})\|_{\ell_1} &\leq \sum_{s \in S} \left| \int_{N(s) \setminus \{s\}} \epsilon(s) k(t-s) h(dt) \right| + \sum_{s \in S} \left| \int_{\mathcal{N}(s)} u_s(t) h(dt) \right| + \left| \int_{\mathcal{F}} v(t) h(dt) \right|, \\ &\leq \sum_{s \in S} \int_{N(s) \setminus \{s\}} \left(1 - \frac{1}{2\lambda_c^2} |k_n''(\gamma_c)| (t-s)^2 \right) |h|(dt) + \sum_{s \in S} \left| \int_{\mathcal{N}(s)} u_s(t) h(dt) \right| \\ &\quad + (1-\beta) \int_{\mathcal{F}} |h|(dt), \\ &\leq \|\mathcal{P}_{\mathcal{N} \setminus S}(h)\|_{\ell_1} - |k_n''(\gamma_c)| \mathcal{I}_{\mathcal{N}}^{(2)} + \sum_{s \in S} \left| \int_{\mathcal{N}(s)} u_s(t) h(dt) \right| + (1-\beta) \|\mathcal{P}_{\mathcal{F}}(h)\|_{\ell_1}. \end{aligned}$$

We now focus on bounding the third term. For this purpose we linearize $u_s(t)$ in the interval $\mathcal{N}(s)$. Based on a generalization of the mean value theorem we have that for each $t \in \mathcal{N}(s)$ there exists $\tau \in \mathcal{N}(s)$ such that

$$u_s(t) = u_s(s) + u'_s(s)(t-s) + \frac{u''_s(\tau)}{2}(t-s)^2.$$

We make use of the following lemma

Lemma 19.4.6 *For $t \in \mathcal{N}(s)$ we have*

$$\begin{aligned} |u_s(s)| &\leq u_0 := 2f_0(2OSR, 0), \\ |u'_s(s)| &\leq \frac{u_1}{\lambda_c} := \frac{2}{\lambda_c} k_u^{(1)}(OSR) + \frac{1}{\lambda_c} (2\pi f_0(2OSR, -2OSR) + 8f_1(2OSR, -2OSR)), \\ \max_{t \in \mathcal{N}(s)} |u''_s(t)| &\leq \frac{u_2}{\lambda_c^2} := \frac{1}{\lambda_c^2} \left(\sum_{r=1}^4 k_u^{(2)}(rOSR - \gamma_c) + \sum_{r=1}^4 k_u^{(2)}(rOSR) \right) \\ &\quad + \left(\pi^2 f_0(2OSR, 2\gamma_c - 10OSR) + 8\pi f_1(2OSR, 2\gamma_c - 10OSR) \right. \\ &\quad \left. - 24f_2(2OSR, 2\gamma_c - 10OSR) - 32f_3(2OSR, 2\gamma_c - 10OSR) \right) \\ &\quad + \left(\pi^2 f_0(2OSR, -10OSR) + 8\pi f_1(2OSR, -10OSR) \right. \\ &\quad \left. - 24f_2(2OSR, -10OSR) - 32f_3(2OSR, -10OSR) \right). \end{aligned}$$

Applying the above lemma we have

$$\left| \int_{t \in \mathcal{N}(s)} u_s(t) h(dt) \right| \leq u_0 \left| \int_{t \in \mathcal{N}(s)} h(dt) \right| + u_1 \left| \int_{t \in \mathcal{N}(s)} \frac{(t-s)}{\lambda_c} h(dt) \right| + u_2 \mathcal{I}_{\mathcal{N}(s)}^{(2)}.$$

Therefore, summing over $s \in S$ we have

$$\sum_{s \in S} \left| \int_{t \in \mathcal{N}(s)} u_s(t) h(dt) \right| \leq u_0 \mathcal{I}_{\mathcal{N}}^{(0)} + u_1 \mathcal{I}_{\mathcal{N}}^{(1)} + u_2 \mathcal{I}_{\mathcal{N}}^{(2)}.$$

Finally, applying Lemma 19.4.7 below we arrive at

$$\|\mathcal{P}_S(h)\|_{\ell_1} \leq \|\mathcal{P}_{\mathcal{N} \setminus S}(h)\|_{\ell_1} - C \mathcal{I}_{\mathcal{N}}^{(2)} + (1 - D) \|\mathcal{P}_{\mathcal{F}}(h)\|_{\ell_1},$$

where

$$\begin{aligned}
 C(OSR, \gamma_c, \lambda_c) &= |k_n''(\gamma_c)| - u_2 \\
 &\quad - \frac{u_0 p_2}{1 - p_0 - \lambda_c(q_1 + p_1 q_0 - p_0 q_1)} \\
 &\quad - \lambda_c \frac{u_1(q_2 + q_0 p_2 - p_0 q_2) - u_0(q_1 p_2 - p_1 q_2)}{1 - p_0 - \lambda_c(q_1 + p_1 q_0 - p_0 q_1)} \\
 D(OSR, \gamma_c, \lambda_c) &= \beta - \frac{u_0(1 - \beta)}{1 - p_0 - \lambda_c(q_1 + p_1 q_0 - p_0 q_1)} \\
 &\quad + \lambda_c \frac{u_0(q_1 - p_1 \beta' - q_1 \beta) - u_1(q_0 - q_0 \beta + (1 - p_0) \beta')}{1 - p_0 - \lambda_c(q_1 + p_1 q_0 - p_0 q_1)}
 \end{aligned}$$

All variables above are a function of λ_c, γ_c, OSR , and will be defined in the proof section.

Lemma 19.4.7 *We have*

$$\begin{aligned}
 \mathcal{I}_N^{(0)} &\leq \frac{p_2 - \lambda_c(q_1 p_2 - p_1 q_2)}{1 - p_0 - \lambda_c(q_1 + p_1 q_0 - p_0 q_1)} \mathcal{I}_N^{(2)} + \frac{1 - \beta - \lambda_c(q_1 - p_1 \beta' - q_1 \beta)}{1 - p_0 - \lambda_c(q_1 + p_1 q_0 - p_0 q_1)} \|\mathcal{P}_{\mathcal{F}}(h)\|_{\ell_1}, \\
 \mathcal{I}_N^{(1)} &\leq \frac{\lambda_c(q_2 + q_0 p_2 - p_0 q_2)}{1 - p_0 - \lambda_c(q_1 + p_1 q_0 - p_0 q_1)} \mathcal{I}_N^{(2)} + \frac{\lambda_c(q_0 - q_0 \beta + (1 - p_0) \beta')}{1 - p_0 - \lambda_c(q_1 + p_1 q_0 - p_0 q_1)} \|\mathcal{P}_{\mathcal{F}}(h)\|_{\ell_1}.
 \end{aligned}$$

By showing that $C > 0$ and $D > 0$ we can conclude that $\|\mathcal{P}_S(h)\|_{\ell_1} < \|\mathcal{P}_{S^c}(h)\|_{\ell_1}$. We use $\gamma_c = 0.4473$. For these choices we draw C and D as a function of OSR and Ω and see where it crosses zero.

19.4.3 Proof of main lemmas

First we recall two results on infinite series. We define

$$\begin{aligned}
 f_0(a, b) &= \frac{1}{2a} \left(\psi^{(0)}\left(1 - \frac{b}{a} + \frac{1}{a}\right) - \psi^{(0)}\left(1 - \frac{b}{a} - \frac{1}{a}\right) \right), \\
 f_1(a, b) &= -\frac{1}{4a^2} \left(\psi^{(1)}\left(1 - \frac{b}{a} + \frac{1}{a}\right) - \psi^{(1)}\left(1 - \frac{b}{a} - \frac{1}{a}\right) \right), \\
 f_2(a, b) &= -\frac{1}{2}f_0(a, b) + \frac{1}{4a^2} \left(\psi^{(1)}\left(1 - \frac{b}{a} + \frac{1}{a}\right) + \psi^{(1)}\left(1 - \frac{b}{a} - \frac{1}{a}\right) \right), \\
 f_3(a, b) &= \frac{3}{8}f_0(a, b) - \frac{3}{16a^2} \left(\psi^{(1)}\left(1 - \frac{b}{a} + \frac{1}{a}\right) + \psi^{(1)}\left(1 - \frac{b}{a} - \frac{1}{a}\right) \right) \\
 &\quad + \frac{1}{16a^3} \left(\psi^{(2)}\left(1 - \frac{b}{a} + \frac{1}{a}\right) - \psi^{(2)}\left(1 - \frac{b}{a} - \frac{1}{a}\right) \right).
 \end{aligned}$$

Lemma 19.4.8 *We have*

$$\begin{aligned}
 \sum_{r=1}^{\infty} \frac{1}{(ar-b)^2 - 1} &= f_0(a, b), \\
 \sum_{r=1}^{\infty} \frac{ar-b}{((ar-b)^2 - 1)^2} &= f_1(a, b), \\
 \sum_{r=1}^{\infty} \frac{1}{((ar-b)^2 - 1)^2} &= f_2(a, b), \\
 \sum_{r=1}^{\infty} \frac{1}{((ar-b)^2 - 1)^3} &= f_3(a, b).
 \end{aligned}$$

19.4.3.1 Proof of Lemma 19.4.6

We have

$$\begin{aligned} |u_s(s)| &= \left| \sum_{s' \in S \setminus \{s\}} \epsilon(s') k(s - s') \right| \\ &\leq 2 \sum_{r=1}^{\infty} \frac{1}{4r^2 OSR^2 - 1} \\ &= 2f_0(2OSR, 0). \end{aligned}$$

Define

$$k_u^{(1)}(\gamma) = \begin{cases} 0.944056, & |\gamma| \leq 1.25779 \\ \frac{\pi}{4\gamma^2-1} + \frac{8|\gamma|}{(4\gamma^2-1)^2}, & |\gamma| > 1.25779 \end{cases}.$$

For the derivative we have

$$\begin{aligned} |u'_s(s)| &= \left| \sum_{s' \in S \setminus \{s\}} \epsilon(s') k'(s - s') \right| \\ &\leq \sum_{s' \in S \setminus \{s\}} |k'(s - s')| \\ &\leq \frac{2}{\lambda_c} \sum_{r=1}^{\infty} k_u^{(1)}(rOSR) \\ &= \frac{2}{\lambda_c} k_u^{(1)}(OSR) + \frac{2}{\lambda_c} \sum_{r=1}^{\infty} k_u^{(1)}(rOSR + OSR) \\ &\leq \frac{2}{\lambda_c} k_u^{(1)}(OSR) + \frac{1}{\lambda_c} \left(2\pi \sum_{r=1}^{\infty} \frac{1}{(2rOSR + 2OSR)^2 - 1} + 8 \sum_{r=1}^{\infty} \frac{2rOSR + 2OSR}{((2rOSR + 2OSR)^2 - 1)^2} \right) \\ &= \frac{2}{\lambda_c} k_u^{(1)}(OSR) + \frac{1}{\lambda_c} (2\pi f_0(2OSR, -2OSR) + 8f_1(2OSR, -2OSR)). \end{aligned}$$

Finally, for the second derivative we have

$$u''_s(t) = \sum_{s' \in S \setminus \{s\}} \epsilon(s') k''(t - s')$$

Define

$$k_u^{(2)}(\gamma) = \begin{cases} k_n''(\gamma), & |\gamma| \leq 0.442 \\ 1.1835, & 0.442 < |\gamma| \leq 1.4565 \\ k_n''(\gamma), & 1.4565 < |\gamma| \leq 2.1136 \\ 0.3325, & 2.1136 < |\gamma| \leq 2.7656 \\ k_n''(\gamma), & 2.7656 < |\gamma| \leq 3.1417 \\ 0.1704, & 3.1417 < |\gamma| \leq 4.31 \\ \frac{16\pi|\gamma|}{(4\gamma^2-1)^2} + \frac{\pi^2(4\gamma^2-1)^2 - 8(12\gamma^2+1)}{(4\gamma^2-1)^3}, & |\gamma| > 4.31 \end{cases}.$$

Note that we used the fact that for $\gamma \geq \frac{\sqrt{\pi^2 + 4(3 + \sqrt{9 + 2\pi^2})}}{2\pi}$ we have

$$\left| \pi^2(4\gamma^2 - 1)^2 - 8(12\gamma^2 + 1) \right| \leq \pi^2(4\gamma^2 - 1)^2 - 8(12\gamma^2 + 1)$$

Therefore we have

$$|k''(t)| \leq \frac{1}{\lambda_c^2} k_u^{(2)}(\gamma).$$

Note that $k_u^{(2)}(\gamma)$ is decreasing for $\gamma \geq 0$. We use the change of variables $t = s + \gamma\lambda_c$. For $t \in \mathcal{N}$ we have the following chain of inequalities

$$\begin{aligned}
|u_s''(t)| &\leq \left| \sum_{s' \in S \setminus \{s\}} \epsilon(s') k''(s - s') \right| \\
&\leq \frac{1}{\lambda_c^2} \sum_{r=1}^{\infty} k_u^{(2)}(rOSR - |\gamma|) + \frac{1}{\lambda_c^2} \sum_{r=1}^{\infty} k_u^{(2)}(rOSR + |\gamma|) \\
&\leq \frac{1}{\lambda_c^2} \sum_{r=1}^{\infty} k_u^{(2)}(rOSR - \gamma_c) + \frac{1}{\lambda_c^2} \sum_{r=1}^{\infty} k_u^{(2)}(rOSR) \\
&= \frac{1}{\lambda_c^2} \left(\sum_{r=1}^4 k_u^{(2)}(rOSR - \gamma_c) + \sum_{r=1}^4 k_u^{(2)}(rOSR) \right) \\
&\quad + \frac{1}{\lambda_c^2} \sum_{r=1}^{\infty} k_u^{(2)}(rOSR + 5OSR - \gamma_c) + \frac{1}{\lambda_c^2} \sum_{r=1}^{\infty} k_u^{(2)}(rOSR + 5OSR) \\
&= \frac{1}{\lambda_c^2} \left(\sum_{r=1}^4 k_u^{(2)}(rOSR - \gamma_c) + \sum_{r=1}^4 k_u^{(2)}(rOSR) \right) \\
&\quad + \frac{1}{\lambda_c^2} \left(8\pi \sum_{r=1}^{\infty} \frac{(2rOSR + 10OSR - 2\gamma_c)}{((2rOSR + 10OSR - 2\gamma_c)^2 - 1)^2} + \pi^2 \sum_{r=1}^{\infty} \frac{1}{(2rOSR + 10OSR - 2\gamma_c)^2 - 1} \right. \\
&\quad \left. - 24 \sum_{r=1}^{\infty} \frac{1}{((2rOSR + 10OSR - 2\gamma_c)^2 - 1)^2} - 32 \sum_{r=1}^{\infty} \frac{1}{((2rOSR + 10OSR - 2\gamma_c)^2 - 1)^3} \right) \\
&\quad + \frac{1}{\lambda_c^2} \left(8\pi \sum_{r=1}^{\infty} \frac{2rOSR + 10OSR}{((2rOSR + 10OSR)^2 - 1)^2} + \pi^2 \sum_{r=1}^{\infty} \frac{1}{(2rOSR + 10OSR)^2 - 1} \right. \\
&\quad \left. - 24 \sum_{r=1}^{\infty} \frac{1}{((2rOSR + 10OSR)^2 - 1)^2} - 32 \sum_{r=1}^{\infty} \frac{1}{((2rOSR + 10OSR)^2 - 1)^3} \right), \\
&= \frac{1}{\lambda_c^2} \left(\sum_{r=1}^4 k_u^{(2)}(rOSR - \gamma_c) + \sum_{r=1}^4 k_u^{(2)}(rOSR) \right) \\
&\quad + \frac{1}{\lambda_c^2} \left(\pi^2 f_0(2OSR, 2\gamma_c - 10OSR) + 8\pi f_1(2OSR, 2\gamma_c - 10OSR) \right. \\
&\quad \left. - 24f_2(2OSR, 2\gamma_c - 10OSR) - 32f_3(2OSR, 2\gamma_c - 10OSR) \right) \\
&\quad + \frac{1}{\lambda_c^2} \left(\pi^2 f_0(2OSR, -10OSR) + 8\pi f_1(2OSR, -10OSR) \right. \\
&\quad \left. - 24f_2(2OSR, -10OSR) - 32f_3(2OSR, -10OSR) \right).
\end{aligned}$$

19.4.3.2 Proof of Lemma 19.4.7

For $s \in S$

$$\begin{aligned}\epsilon_0(s) &= \frac{\overline{\mathcal{I}_{\mathcal{N}(s)}^{(0)}}}{|\mathcal{I}_{\mathcal{N}(s)}^{(0)}|} \\ \epsilon_1(s) &= \frac{\overline{\mathcal{I}_{\mathcal{N}(s)}^{(1)}}}{|\mathcal{I}_{\mathcal{N}(s)}^{(1)}|}.\end{aligned}$$

Also, define

$$\begin{aligned}p(t) &= \sum_{s \in S} \epsilon_0(s)k(t-s), \\ p_s(t) &= \epsilon_0(s) - p(t), \\ q(t) &= \sum_{s \in S} \epsilon_1(s)g(t-s), \\ q_s(t) &= \epsilon_1(s)(t-s) - q(t).\end{aligned}$$

Now note that

$$\begin{aligned}\mathcal{I}_{\mathcal{N}(s)}^{(0)} &= \int_{\mathcal{N}(s)} \epsilon_0(s)h(dt) \\ &= \int_{\mathcal{N}(s)} p(t)h(dt) + \int_{\mathcal{N}(s)} (\epsilon_0(s) - p(t))h(dt)\end{aligned}$$

Summing over $s \in S$

$$\begin{aligned}\mathcal{I}_{\mathcal{N}}^{(0)} &= \sum_{s \in S} \mathcal{I}_{\mathcal{N}(s)}^{(0)}, \\ &= \sum_{s \in S} \int_{\mathcal{N}(s)} (\epsilon_0(s) - p(t))h(dt) + \sum_{s \in S} \int_{\mathcal{N}(s)} p(t)h(dt), \\ &= \sum_{s \in S} \int_{\mathcal{N}(s)} p_s(t)h(dt) - \int_{\mathcal{F}} p(t)h(dt), \\ &\leq \sum_{s \in S} \left| \int_{\mathcal{N}(s)} p_s(t)h(dt) \right| + (1-\beta) \|\mathcal{P}_{\mathcal{F}}(h)\|_{\ell_1}. \tag{19.4.2}\end{aligned}$$

Based on a generalization of the mean value theorem we have that for each $t \in \mathcal{N}(s)$ there exists $\tau \in \mathcal{N}(s)$ such that

$$p_s(t) = p_s(s) + p'_s(s)(t - s) + \frac{p''_s(\tau)}{2}(t - s)^2.$$

We make use of the following lemma

Lemma 19.4.9 *If $OSR \geq 1$, for $t \in \mathcal{N}(s)$ we have*

$$\begin{aligned} |p_s(s)| &\leq p_0 := u_0, \\ |p'_s(s)| &\leq \frac{p_1}{\lambda_c} := \frac{u_1}{\lambda_c}, \\ \max_{t \in \mathcal{N}(s)} |p''_s(t)| &\leq \frac{p_2}{\lambda_c^2} := \frac{\pi^2 - 8}{\lambda_c^2} + \frac{u_2}{\lambda_c^2}. \end{aligned}$$

Applying the above lemma we have

$$\left| \int_{t \in \mathcal{N}(s)} p_s(t) h(dt) \right| \leq p_0 \left| \int_{t \in \mathcal{N}(s)} h(dt) \right| + p_1 \left| \int_{t \in \mathcal{N}(s)} \frac{(t-s)}{\lambda_c} h(dt) \right| + p_2 \mathcal{I}_{\mathcal{N}(s)}^{(2)}.$$

Therefore, summing over $s \in S$

$$\sum_{s \in S} \left| \int_{t \in \mathcal{N}(s)} p_s(t) h(dt) \right| \leq p_0 \mathcal{I}_{\mathcal{N}}^{(0)} + p_1 \mathcal{I}_{\mathcal{N}}^{(1)} + p_2 \mathcal{I}_{\mathcal{N}}^{(2)}.$$

Now plugging the latter into (19.4.2) we arrive at

$$\mathcal{I}_{\mathcal{N}}^{(0)} \leq p_0 \mathcal{I}_{\mathcal{N}}^{(0)} + p_1 \mathcal{I}_{\mathcal{N}}^{(1)} + p_2 \mathcal{I}_{\mathcal{N}}^{(2)} + (1 - \beta) \|\mathcal{P}_{\mathcal{F}}(h)\|_{\ell_1}. \quad (19.4.3)$$

Now note that

$$\begin{aligned} \mathcal{I}_{\mathcal{N}(s)}^{(1)} &= \int_{\mathcal{N}(s)} \epsilon_1(s)(t-s) h(dt) \\ &= \int_{\mathcal{N}(s)} q(t) h(dt) + \int_{\mathcal{N}(s)} (\epsilon_1(s)(t-s) - q(t)) h(dt) \end{aligned}$$

Summing over $s \in S$

$$\begin{aligned}
\mathcal{I}_{\mathcal{N}}^{(1)} &= \sum_{s \in S} \mathcal{I}_{\mathcal{N}(s)}^{(1)}, \\
&= \sum_{s \in S} \int_{\mathcal{N}(s)} (\epsilon_1(s)(t-s) - q(t)) h(dt) + \sum_{s \in S} \int_{\mathcal{N}(s)} q(t) h(dt), \\
&= \sum_{s \in S} \int_{\mathcal{N}(s)} q_s(t) h(dt) - \int_{\mathcal{F}} q(t) h(dt), \\
&\leq \sum_{s \in S} \left| \int_{\mathcal{N}(s)} q_s(t) h(dt) \right| + \lambda_c \beta' \|\mathcal{P}_{\mathcal{F}}(h)\|_{\ell_1}.
\end{aligned} \tag{19.4.4}$$

The last line follows from

Lemma 19.4.10

$$\max_{t \in \mathcal{F}} |q(t)| \leq \lambda_c \beta',$$

where

$$\beta' = g_u^{(0)}(\gamma_c) + g_u^{(0)}(OSR/2) + g_u^{(0)}(OSR + \gamma_c) + \frac{1}{\pi} f_0(OSR, -OSR/2) + \frac{1}{\pi} f_0(OSR, -OSR - \gamma_c).$$

Based on a generalization of the mean value theorem we have that for each $t \in \mathcal{N}(s)$ there exists $\tau \in \mathcal{N}(s)$ such that

$$q_s(t) = q_s(s) + q'_s(s)(t-s) + \frac{q''_s(\tau)}{2}(t-s)^2.$$

We make use of the following lemma

Lemma 19.4.11 If $\gamma_c \leq 0.71$, for $t \in \mathcal{N}(s)$ we have

$$\begin{aligned}
|q_s(s)| &\leq \lambda_c q_0 := 2\lambda_c g_u^{(0)}(OSR) + 2\lambda_c \frac{1}{\pi} f_0(OSR, -OSR), \\
|q'_s(s)| &\leq q_1 := 2g_u^{(1)}(OSR) + 2f_0(OSR, -OSR) + \frac{4}{\pi} f_1(OSR, -OSR), \\
\max_{t \in \mathcal{N}(s)} |q''_s(t)| &\leq \frac{q_2}{\lambda_c} := \frac{1}{\lambda_c} |g_n''(\gamma_c)| \\
&\quad + \frac{1}{\lambda_c} \left(g_u^{(2)}(OSR - \gamma_c) + g_u^{(2)}(2OSR - \gamma_c) + g_u^{(2)}(3OSR - \gamma_c) \right. \\
&\quad \left. + g_u^{(2)}(OSR) + g_u^{(2)}(2OSR) \right) \\
&\quad + \frac{1}{\lambda_c} \left(\pi f_0(OSR, \gamma_c - 3OSR) + 4f_1(OSR, \gamma_c - 3OSR) \right. \\
&\quad \left. - \frac{6}{\pi} f_2(OSR, \gamma_c - 3OSR) - \frac{8}{\pi} f_3(OSR, \gamma_c - 3OSR) \right) \\
&\quad + \frac{1}{\lambda_c} \left(\pi f_0(OSR, -2OSR) + 4f_1(OSR, -2OSR) \right. \\
&\quad \left. - \frac{6}{\pi} f_2(OSR, -2OSR) - \frac{8}{\pi} f_3(OSR, -2OSR) \right)
\end{aligned}$$

Applying the above lemma we have

$$\left| \int_{t \in \mathcal{N}(s)} q_s(t) h(dt) \right| \leq \lambda_c q_0 \left| \int_{t \in \mathcal{N}(s)} h(dt) \right| + \lambda_c q_1 \left| \int_{t \in \mathcal{N}(s)} \frac{(t-s)}{\lambda_c} h(dt) \right| + \lambda_c q_2 \mathcal{I}_{\mathcal{N}(s)}^{(2)}.$$

Therefore, summing over $s \in S$ and plugging the latter into (19.4.4) we arrive at

$$\mathcal{I}_{\mathcal{N}}^{(1)} \leq \lambda_c q_0 \mathcal{I}_{\mathcal{N}}^{(0)} + \lambda_c q_1 \mathcal{I}_{\mathcal{N}}^{(1)} + \lambda_c q_2 \mathcal{I}_{\mathcal{N}}^{(2)} + \lambda_c \beta' \|\mathcal{P}_{\mathcal{F}}(h)\|_{\ell_1}. \quad (19.4.5)$$

Now notice that

$$\begin{cases} ax \leq by + A \\ dy \leq cx + B \end{cases} \xrightarrow{ad-bc>0} \begin{cases} x \leq \frac{1}{(ad-bc)}(dA + bB) \\ y \leq \frac{1}{(ad-bc)}(cA + aB) \end{cases}$$

Applying the latter identity to (19.4.3) and (19.4.5) with the following choices we arrive at the result.

$$\begin{aligned} a &= 1 - p_0, \quad b = p_1, \quad c = \lambda_c q_0, \quad d = 1 - \lambda_c q_1, \\ A &= p_2 \mathcal{I}_{\mathcal{N}}^{(2)} + (1 - \beta) \|\mathcal{P}_{\mathcal{F}}(h)\|_{\ell_1}, \\ B &= \lambda_c q_2 \mathcal{I}_{\mathcal{N}}^{(2)} + \lambda_c \beta' \|\mathcal{P}_{\mathcal{F}}(h)\|_{\ell_1}. \end{aligned}$$

19.4.3.3 Proof of Lemma 19.4.4

We assume that $s = 0$ and we work with the normalization $\gamma = 2\Omega t$. We would need to show the identity for $\gamma_c \leq \gamma \leq \gamma_+/2$ where γ_+ is the closest spike to 0. We define

$$k_u^{(0)}(\gamma) = \begin{cases} \frac{\cos(\pi\gamma)}{4\gamma^2-1}, & 0 \leq \gamma \leq 1 \\ \frac{1}{4\gamma^2-1}, & \gamma > 1 \end{cases}.$$

which is always decreasing. Therefore, it is easy to see that this is equivalent to controlling the function right-hand side of the expression below for $\gamma_c \leq \gamma \leq OSR/2$. That is

$$\begin{aligned} \max_{t \in \mathcal{F}} |v(t)| &\leq k_u^{(0)}(\gamma) + \sum_{r=1}^{\infty} k_u^{(0)}(rOSR - \gamma) + \sum_{r=1}^{\infty} k_u^{(0)}(rOSR + \gamma) \\ &\leq k_u^{(0)}(\gamma_c) + \sum_{r=1}^{\infty} k_u^{(0)}(rOSR - OSR/2) + \sum_{r=1}^{\infty} k_u^{(0)}(rOSR + \gamma_c) \\ &= k_u^{(0)}(\gamma_c) + k_u^{(0)}(OSR/2) + \sum_{r=1}^{\infty} k_u^{(0)}(rOSR + OSR/2) + \sum_{r=1}^{\infty} k_u^{(0)}(rOSR + \gamma_c) \\ &\leq k_u^{(0)}(\gamma_c) + k_u^{(0)}(OSR/2) + \sum_{r=1}^{\infty} \frac{1}{(2rOSR + OSR)^2 - 1} + \sum_{r=1}^{\infty} \frac{1}{(2rOSR + 2\gamma_c)^2 - 1} \\ &= k_u^{(0)}(\gamma_c) + k_u^{(0)}(OSR/2) + f_0(2OSR, -OSR) + f_0(2OSR, -2\gamma_c). \end{aligned}$$

The next to the last line (inequality) holds as long as $OSR \geq 1$.

19.4.3.4 Proof of Lemma 19.4.9

The proof of the first two parts are the same as Lemma 19.4.6. We proceed with the proof of the last part.

$$p_s''(t) = -\epsilon_0(s)k''(t-s) + \sum_{s' \in S \setminus \{s\}} \epsilon_0(s')k''(s-s')$$

We use the change of variables $t = s + \gamma\lambda_c$ we have for $t \in \mathcal{N}$ and use a similar argument as in Lemma 19.4.6.

$$\begin{aligned} |p_s''(t)| &\leq |k''(t-s)| + \left| \sum_{s' \in S \setminus \{s\}} \epsilon(s')k''(s-s') \right| \\ &\leq \frac{1}{\lambda_c^2} |k_n''(\gamma)| + \frac{u_2}{\lambda_c^2} \\ &\leq \frac{\pi^2 - 8 + u_2}{\lambda_c^2}. \end{aligned}$$

19.4.3.5 Proof of Lemma 19.4.10

We assume that $s = 0$ and we work with the normalization $\gamma = 2\Omega t$. We would need to show the identity for $\gamma_c \leq \gamma \leq \gamma_+/2$ where γ_+ is the closest spike to 0. We define

$$g_u^{(0)}(\gamma) = \begin{cases} 0.520885, & |\gamma| \leq 0.837472 \\ \frac{1}{\pi} \frac{\sin(\pi\gamma)}{\gamma^2-1}, & 0.837472 \leq |\gamma| \leq 1.5 \\ \frac{1}{\pi} \frac{1}{\gamma^2-1}, & |\gamma| > 1.5 \end{cases}$$

which is always non-increasing. Therefore, it is easy to see that this is equivalent to controlling the function right-hand side of the expression below for $\gamma_c \leq \gamma \leq OSR/2$.

$$\begin{aligned}
\max_{t \in \mathcal{F}} |q(t)| &\leq \lambda_c g_u^{(0)}(\gamma) + \lambda_c \sum_{r=1}^{\infty} g_u^{(0)}(rOSR - \gamma) + \lambda_c \sum_{r=1}^{\infty} g_u^{(0)}(rOSR + \gamma) \\
&\leq \lambda_c g_u^{(0)}(\gamma_c) + \lambda_c \sum_{r=1}^{\infty} g_u^{(0)}(rOSR - OSR/2) + \lambda_c \sum_{r=1}^{\infty} g_u^{(0)}(rOSR + \gamma_c) \\
&= \lambda_c g_u^{(0)}(\gamma_c) + \lambda_c g_u^{(0)}(OSR/2) + \lambda_c g_u^{(0)}(OSR + \gamma_c) + \lambda_c \sum_{r=1}^{\infty} g_u^{(0)}(rOSR + OSR/2) \\
&\quad + \lambda_c \sum_{r=1}^{\infty} g_u^{(0)}(rOSR + OSR + \gamma_c) \\
&\leq \lambda_c g_u^{(0)}(\gamma_c) + \lambda_c g_u^{(0)}(OSR/2) + \lambda_c g_u^{(0)}(OSR + \gamma_c) + \lambda_c \frac{1}{\pi} \sum_{r=1}^{\infty} \frac{1}{(rOSR + OSR/2)^2 - 1} \\
&\quad + \lambda_c \frac{1}{\pi} \sum_{r=1}^{\infty} \frac{1}{(rOSR + OSR + \gamma_c)^2 - 1} \\
&= \lambda_c g_u^{(0)}(\gamma_c) + \lambda_c g_u^{(0)}(OSR/2) + \lambda_c g_u^{(0)}(OSR + \gamma_c) + \lambda_c \frac{1}{\pi} f_0(OSR, -OSR/2) \\
&\quad + \lambda_c \frac{1}{\pi} f_0(OSR, -OSR - \gamma_c).
\end{aligned}$$

19.4.3.6 Proof of Lemma 19.4.11

We have

$$\begin{aligned}
|q_s(s)| &= \left| \sum_{s' \in S \setminus \{s\}} \epsilon_1(s') g(s - s') \right| \\
&\leq 2\lambda_c \sum_{r=1}^{\infty} \left| g_u^{(0)}(rOSR) \right| \\
&\leq 2\lambda_c g_u^{(0)}(OSR) + 2\lambda_c \sum_{r=1}^{\infty} \left| g_u^{(0)}(rOSR + OSR) \right| \\
&\leq 2\lambda_c g_u^{(0)}(OSR) + 2\lambda_c \frac{1}{\pi} \sum_{r=1}^{\infty} \frac{1}{(rOSR + OSR)^2 - 1} \\
&= 2\lambda_c g_u^{(0)}(OSR) + 2\lambda_c \frac{1}{\pi} f_0(OSR, -OSR).
\end{aligned}$$

We define

$$g_u^{(1)}(\gamma) = \begin{cases} 1 - (\frac{1}{6} - \frac{1}{9\pi})\gamma^2, & |\gamma| \leq 2 \\ \frac{1}{\gamma^2-1} + \frac{2\gamma}{\pi(\gamma^2-1)^2}, & |\gamma| > 2 \end{cases}$$

For the derivative we have

$$\begin{aligned} |q_s'(s)| &= \left| \sum_{s' \in S \setminus \{s\}} \epsilon(s') g'(s-s') \right| \\ &\leq \sum_{s' \in S \setminus \{s\}} |g'(s-s')| \\ &\leq 2 \sum_{r=1}^{\infty} g_u^{(1)}(rOSR) \\ &\leq 2g_u^{(1)}(rOSR) + 2 \sum_{r=1}^{\infty} g_u^{(1)}(rOSR + OSR) \\ &= 2g_u^{(1)}(rOSR) + 2 \sum_{r=1}^{\infty} \frac{1}{(rOSR + OSR)^2 - 1} + \frac{4}{\pi} \sum_{r=1}^{\infty} \frac{(rOSR + OSR)}{((rOSR + OSR)^2 - 1)^2} \\ &= 2g_u^{(1)}(OSR) + 2f_0(OSR, -OSR) + \frac{4}{\pi} f_1(OSR, -OSR). \end{aligned}$$

We proceed with the proof of the last part.

$$q_s''(t) = -\epsilon_1(s) g''(t-s) + \sum_{s' \in S \setminus \{s\}} \epsilon_1(s') g''(s-s')$$

Define

$$g_u^{(2)}(\gamma) = \begin{cases} 1.798 - 0.1467\gamma^2, & |\gamma| \leq 2.46951 \\ \frac{4|\gamma|}{(\gamma^2-1)^2} + \frac{\pi}{\gamma^2-1} - \frac{6\gamma^2+2}{\pi(\gamma^2-1)^3}, & |\gamma| \geq 2.46951 \end{cases}$$

We use the change of variables $t = s + \gamma\lambda_c$ we have for $t \in \mathcal{N}$

$$\begin{aligned}
|q_s''(t)| &\leq |g''(t-s)| + \left| \sum_{s' \in S \setminus \{s\}} \epsilon_1(s') g''(s-s') \right| \\
&\leq |g''(t-s)| + \sum_{s' \in S \setminus \{s\}} |g''(s-s')| \\
&\leq \frac{1}{\lambda_c^2} |g_n''(\gamma)| + \frac{1}{\lambda_c^2} \sum_{r=1}^{\infty} \left| g_u^{(2)}(rOSR - |\gamma|) \right| + \frac{1}{\lambda_c^2} \sum_{r=1}^{\infty} \left| g_u^{(2)}(rOSR + |\gamma|) \right| \\
&\leq \frac{1}{\lambda_c^2} |g_n''(\gamma_c)| + \frac{1}{\lambda_c^2} \sum_{r=1}^{\infty} g_u^{(2)}(rOSR - \gamma_c) + \frac{1}{\lambda_c^2} \sum_{r=1}^{\infty} g_u^{(2)}(rOSR) \\
&= \frac{1}{\lambda_c^2} |g_n''(\gamma_c)| + \frac{1}{\lambda_c^2} \left(g_u^{(2)}(OSR - \gamma_c) + g_u^{(2)}(2OSR - \gamma_c) + g_u^{(2)}(3OSR - \gamma_c) \right. \\
&\quad \left. + g_u^{(2)}(OSR) + g_u^{(2)}(2OSR) \right) \\
&\quad + \frac{1}{\lambda_c^2} \sum_{r=1}^{\infty} g_u^{(2)}(rOSR + 3OSR - \gamma_c) + \frac{1}{\lambda_c^2} \sum_{r=1}^{\infty} g_u^{(2)}(rOSR + 2OSR) \\
&= \frac{1}{\lambda_c^2} |g_n''(\gamma_c)| + \frac{1}{\lambda_c^2} \left(g_u^{(2)}(OSR - \gamma_c) + g_u^{(2)}(2OSR - \gamma_c) + g_u^{(2)}(3OSR - \gamma_c) \right. \\
&\quad \left. + g_u^{(2)}(OSR) + g_u^{(2)}(2OSR) \right) \\
&\quad + \frac{1}{\lambda_c^2} \left(\pi \sum_{r=1}^{\infty} \frac{1}{(rOSR + 3OSR - \gamma_c)^2 - 1} - \frac{6}{\pi} \sum_{r=1}^{\infty} \frac{1}{((rOSR + 3OSR - \gamma_c)^2 - 1)^2} \right. \\
&\quad \left. - \frac{8}{\pi} \sum_{r=1}^{\infty} \frac{1}{((rOSR + 3OSR - \gamma_c)^2 - 1)^3} + 4 \sum_{r=1}^{\infty} \frac{(rOSR + 3OSR - \gamma_c)}{((rOSR + 3OSR - \gamma_c)^2 - 1)^2} \right) \\
&\quad + \frac{1}{\lambda_c^2} \left(\pi \sum_{r=1}^{\infty} \frac{1}{(rOSR + 2OSR)^2 - 1} - \frac{6}{\pi} \sum_{r=1}^{\infty} \frac{1}{((rOSR + 2OSR)^2 - 1)^2} \right. \\
&\quad \left. - \frac{8}{\pi} \sum_{r=1}^{\infty} \frac{1}{((rOSR + 2OSR)^2 - 1)^3} + 4 \sum_{r=1}^{\infty} \frac{(rOSR + 2OSR)}{((rOSR + 2OSR)^2 - 1)^2} \right).
\end{aligned}$$

Thus,

$$\begin{aligned}
|q_s''(t)| &\leq \frac{1}{\lambda_c^2} |g_n''(\gamma_c)| + \frac{1}{\lambda_c^2} \left(g_u^{(2)}(OSR - \gamma_c) + g_u^{(2)}(2OSR - \gamma_c) + g_u^{(2)}(3OSR - \gamma_c) \right. \\
&\quad \left. + g_u^{(2)}(OSR) + g_u^{(2)}(2OSR) \right) \\
&+ \frac{1}{\lambda_c^2} \left(\pi f_0(OSR, \gamma_c - 3OSR) + 4f_1(OSR, \gamma_c - 3OSR) \right. \\
&\quad \left. - \frac{6}{\pi} f_2(OSR, \gamma_c - 3OSR) - \frac{8}{\pi} f_3(OSR, \gamma_c - 3OSR) \right) \\
&+ \frac{1}{\lambda_c^2} \left(\pi f_0(OSR, -2OSR) + 4f_1(OSR, -2OSR) \right. \\
&\quad \left. - \frac{6}{\pi} f_2(OSR, -2OSR) - \frac{8}{\pi} f_3(OSR, -2OSR) \right).
\end{aligned}$$

This expression holds in the interval in which $|g_n''(\gamma_c)|$ is increasing ($\gamma_c \leq 0.71$).

Appendix A

Geometric Perspective on the subspace detection property

Our aim in this section is to provide a geometric understanding of the subspace detection property and of the sufficient condition presented in Section 5.3.1.

We have seen that subspace detection property holds if for each point \mathbf{x}_i , the closest face to \mathbf{x}_i resides in the same subspace. To establish a geometric characterization, consider an arbitrary point, for instance $\mathbf{x}_i^{(\ell)} \in S_\ell$ as in Figure A.1. Now construct the symmetrized convex hull of all the other points in S_ℓ indicated by \mathcal{P}_{-i}^ℓ in the figure. Consider the face of \mathcal{P}_{-i}^ℓ that is closest to $\mathbf{x}_i^{(\ell)}$; this face is shown in Figure A.1 by the line segment in red. Also, consider the plane passing through this segment and orthogonal to S_ℓ along with its reflection about the origin; this is shown in Figure A.1 by the light grey planes. Set $R_i^{(\ell)}$ to be the region of space restricted between these two planes. Intuitively, if no two points on the other subspaces lie outside of $R_i^{(\ell)}$, then the face chosen by the algorithm is as in the figure, and lies in S_ℓ .

To illustrate this point further, suppose there are two points not in S_ℓ lying outside of the region $R_i^{(\ell)}$ as in Figure A.2. In this case, the closest face does not lie in S_ℓ as can be seen in the figure. Therefore, one could intuitively argue that a sufficient condition for the closest face to lie in S_ℓ is that the projections onto S_ℓ of the points from all the other subspaces do not lie outside of regions $R_i^{(\ell)}$ for all points $\mathbf{x}_i^{(\ell)}$ in subspace S_ℓ . This condition is closely related to the sufficient condition stated

APPENDIX A. GEOMETRIC PERSPECTIVE ON THE SUBSPACE DETECTION PROPERTY

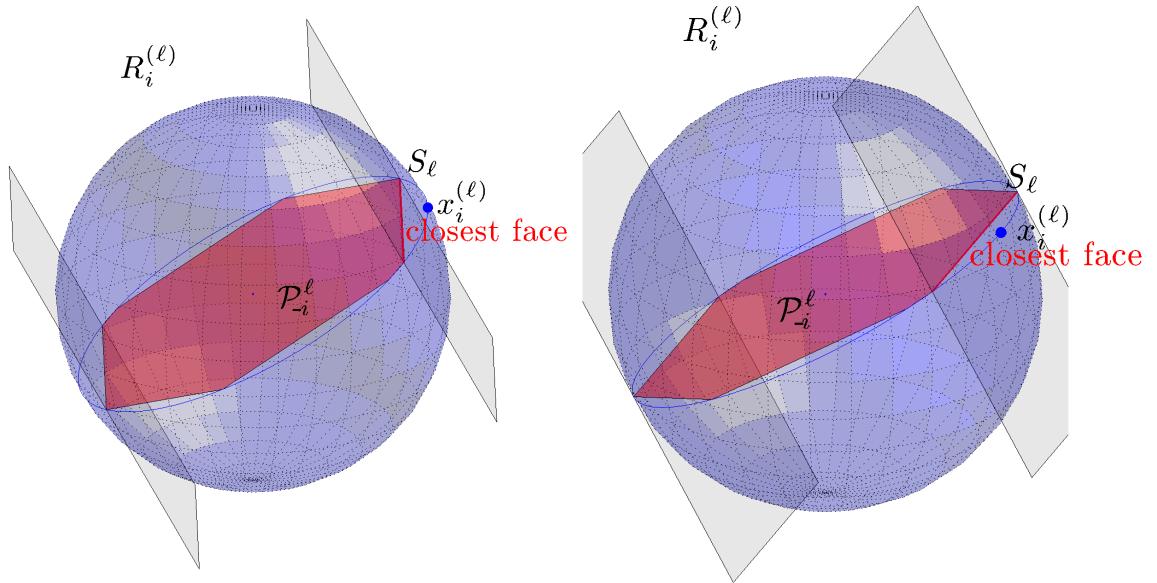


Figure A.1: Illustration of ℓ_1 minimization when the subspace detection property holds. Same object seen from different angles.

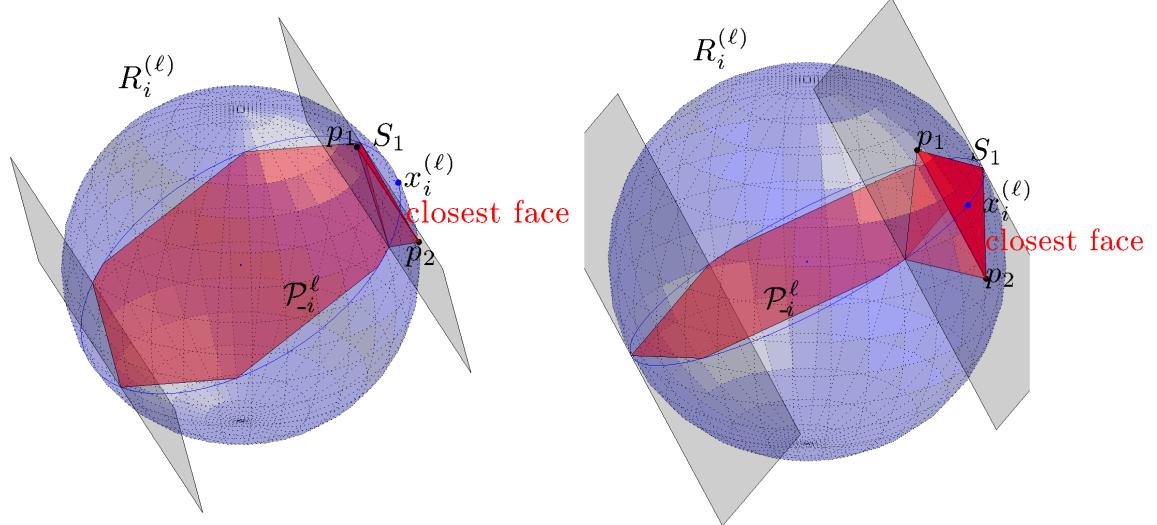


Figure A.2: Illustration of ℓ_1 minimization when the subspace detection property fails. Same object seen from different angles.

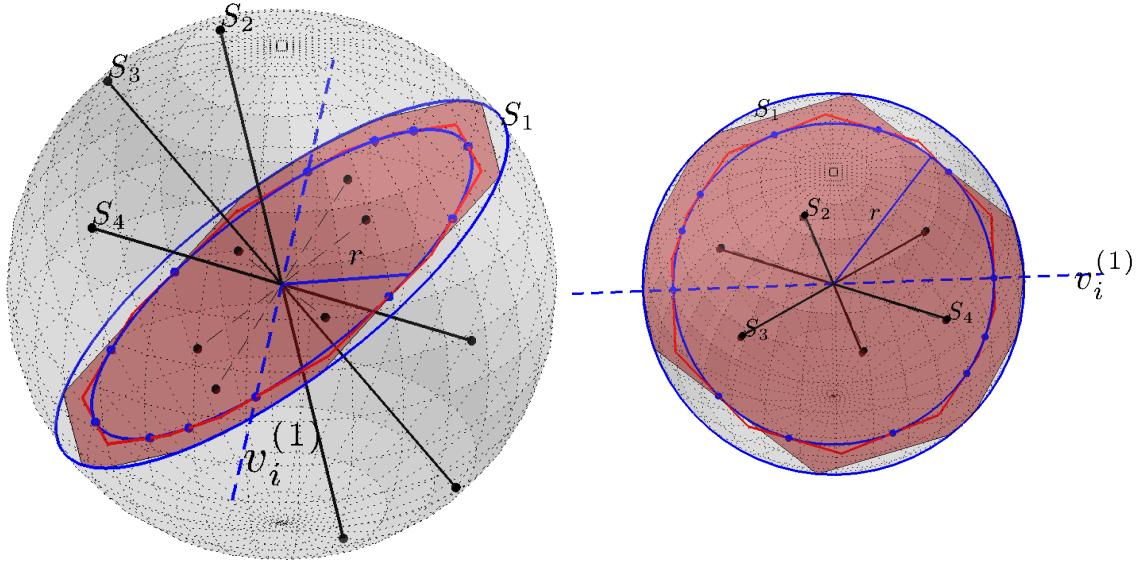


Figure A.3: Geometric view of (5.3.1). The right figure is seen from a direction orthogonal to S_1 .

in Theorem 5.3.5. More, precisely the dual directions $\mathbf{v}_i^{(\ell)}$ approximate the normal directions to the restricting planes $R_i^{(\ell)}$, and $\min_i r(\mathcal{P}_{-i}^{\ell})$ the distance of these planes from the origin.

Finally, to understand the sufficient condition of Theorem 5.3.5, we will use Figure A.3. We focus on a single subspace, say S_1 . As previously stated, a sufficient condition is to have all points not in S_1 to have small coherence with the dual directions of the points in S_1 . The dual directions are depicted in Figure A.3 (blue dots). One such dual direction line is shown as the dashed blue line in the figure. The points that have low coherence with the dual directions are the points whose projection onto subspace S_1 lie inside the red polytope. As can be seen, this polytope approximates the intersection of regions $R_i^{(1)} (\cap_{i=1}^{N_1} R_i^{(1)})$ and subspace S_1 . This helps understanding the difference between the condition imposed by Elhamifar and Vidal and our condition; in this setting, their condition essentially states that the projection of the points on all other subspaces onto subspace S_1 must lie inside the blue circle. By looking at Figure A.3, one might draw the conclusion that these conditions are very similar, i.e. the red polytope and the blue ball restrict almost the same region.

APPENDIX A. GEOMETRIC PERSPECTIVE ON THE SUBSPACE DETECTION PROPERTY

This is not the case, because as the dimension of the subspace S_1 increases most of the volume of the red polytope will be concentrated around its vertices and the ball will only occupy a very small fraction of the total volume of the polytope.

Appendix B

Standard inequalities in probability

This section collects standard inequalities that shall be used throughout. The first concerns tails of chi-square random variables: a chi-square χ_n^2 with n degrees of freedom obeys

$$\mathbb{P}(\chi_n^2 \geq (1 + \epsilon)n) \leq \exp\left(-\frac{(1 - \log 2)}{2} n\epsilon^2\right). \quad (\text{B.0.1})$$

The second concerns the size of the dot product between a fixed vector and Gaussian random vectors.

Lemma B.0.12 *Suppose \mathbf{A} in $\mathbb{R}^{d \times N}$ has iid $\mathcal{N}(0, 1)$ entries and let $\mathbf{z} \in \mathbb{R}^d$ a unit-norm vector. Then*

$$\|\mathbf{A}^T \mathbf{z}\|_{\ell_\infty} \leq 2\sqrt{2 \log N}$$

with probability at least $1 - \frac{2}{N^2}$. (This also applies if \mathbf{z} is a random vector independent from \mathbf{A} .)

Lemma B.0.13 (Sub-Gaussian rows [232]) *Let \mathbf{A} be an $N \times d$ matrix ($N \geq d$) whose rows are independent sub-Gaussian isotropic random vectors in \mathbb{R}^d . Then for every $t \geq 0$,*

$$\sigma_{\min}(\mathbf{A}) \geq \sqrt{N} - C\sqrt{d} - t$$

with probability at least $1 - e^{-ct^2}$. Here, σ_{\min} is the minimum singular value of \mathbf{A} and $C = C_K$, $c = c_K > 0$ depend only on the sub-Gaussian norm $K = \max_i \|\mathbf{A}_i\|_{\Psi_2}$ of the rows (see [232]).

Lemma B.0.14 *With probability at least $1 - e^{-d_1/2}$,*

$$\sigma_{\min}(\mathbf{Y}_{\parallel}^{(1)}) \geq \sqrt{\left(1 + \sigma^2 \frac{d_1}{n}\right)} \left(\sqrt{\frac{N_1}{d_1}} - 2 \right).$$

Proof This is a trivial consequence of Lemma B.0.13 above with $t = \sqrt{d_1}$. ■

Lemma B.0.15 *If σ and d_1 are as in Section 5.4 equation (5.4.3), all the columns in $\mathbf{Y}_{\parallel}^{(1)}$ and \mathbf{y}_{\parallel} have Euclidean norms in $[3/4, 5/4]$ with probability at least $1 - \frac{1}{N^2}$. (For a single column, the probability is at least equal to $1 - \frac{1}{N^3}$.)*

Proof A column of $\mathbf{Y}_{\parallel}^{(1)}$ or \mathbf{y}_{\parallel} is of the form $\mathbf{a} = \mathbf{x} + \mathbf{z}_{\parallel}$ where \mathbf{x} is uniform on the unit sphere of S_1 and $\mathbf{z} \sim \mathcal{N}(0, (\sigma^2/n)\mathbf{I}_n)$. We have

$$\|\mathbf{x}\|_{\ell_2} - \|\mathbf{z}_{\parallel}\|_{\ell_2} \leq \|\mathbf{a}\|_{\ell_2} \leq \|\mathbf{x}\|_{\ell_2} + \|\mathbf{z}_{\parallel}\|_{\ell_2}.$$

The result follows from $\|\mathbf{x}\|_{\ell_2} = 1$ and $\|\mathbf{z}_{\parallel}\|_{\ell_2} \leq \frac{1}{4}$, which holds with high probability. The latter is a consequence of (B.0.1) since $\|\mathbf{z}_{\parallel}\|_{\ell_2}^2$ (properly normalized) is a chi-square with d_1 degrees of freedom and the bounds on σ and d_1 from (5.4.3). ■

Appendix C

Geometric Lemmas

Consider the linear program

$$\mathbf{x}^* \in \arg \min_{\mathbf{x}} \|\mathbf{x}\|_{\ell_1} \quad \text{subject to} \quad \mathbf{y} = \mathbf{A}\mathbf{x}, \quad (\text{C.0.1})$$

and its dual

$$\boldsymbol{\nu}^* \in \arg \max_{\boldsymbol{\nu}} \langle \mathbf{y}, \boldsymbol{\nu} \rangle \quad \text{subject to} \quad \|\mathbf{A}^T \boldsymbol{\nu}\|_{\ell_\infty} \leq 1. \quad (\text{C.0.2})$$

Lemma C.0.16 *Any dual feasible point obeys*

$$\|\boldsymbol{\nu}\|_{\ell_2} \leq \frac{1}{r(\mathcal{P}(\mathbf{A}))}.$$

Proof Put $r = r(\mathcal{P}(\mathbf{A}))$ for short. By definition, there exists \mathbf{x} with $\|\mathbf{x}\|_{\ell_1} \leq 1$ such that $\mathbf{A}\mathbf{x} = r\boldsymbol{\nu}$. Now,

$$r\|\boldsymbol{\nu}\|_{\ell_2} = \langle \mathbf{A}\mathbf{x}, \boldsymbol{\nu} \rangle = \langle \mathbf{x}, \mathbf{A}^T \boldsymbol{\nu} \rangle \leq \|\mathbf{x}\|_{\ell_1} \|\mathbf{A}^T \boldsymbol{\nu}\|_{\ell_\infty} \leq 1.$$

■

Strong duality $\|\mathbf{x}^*\|_{\ell_1} = \langle \mathbf{y}, \boldsymbol{\nu}^* \rangle \leq \|\mathbf{y}\|_{\ell_2} \|\boldsymbol{\nu}^*\|_{\ell_2}$ also gives:

Lemma C.0.17 *Any optimal solution \mathbf{x}^* to (C.0.1) obeys*

$$\|\mathbf{x}^*\|_{\ell_1} \leq \frac{\|\mathbf{y}\|_{\ell_2}}{r(\mathcal{P}(\mathbf{A}))}.$$

Lemma C.0.18 *Assume $\rho_1 = N_1/d_1 \geq \rho^*$. Then*

$$r(\mathcal{P}(\mathbf{Y}_{\parallel}^{(1)})) \geq \frac{3}{16} \sqrt{\frac{\log(N_1/d_1)}{d_1}},$$

with probability at least $1 - \frac{1}{N^2} - e^{-\sqrt{N_1 d_1}}$.

Proof Suppose $\mathbf{A} \in \mathbb{R}^{d \times N}$ has columns chosen uniformly at random from the unit sphere of \mathbb{R}^d with $\rho = N/d \geq \rho_0$. Then based on Lemma 7.2.4 we have

$$\mathbb{P}\left\{r(\mathcal{P}(\mathbf{A})) < \frac{1}{4} \sqrt{\frac{\log(N/d)}{d}}\right\} \leq e^{-\sqrt{Nd}}.$$

The claim in the lemma follows from the lower bound on the Euclidean norm of the columns of $\mathbf{Y}_{\parallel}^{(1)}$ (Lemma B.0.15) together with the fact that they have uniform orientation. \blacksquare

Corollary C.0.19 *With high probability as above, any dual feasible point to (7.3.12) obeys*

$$\|\boldsymbol{\nu}\|_{\ell_2}^2 \leq \frac{256}{9} \frac{d_1}{\log(N_1/d_1)}.$$

Appendix D

Sharpening Lemma 7.3.3 Asymptotically

Here, we assume that the ratio $\rho_1 = N_1/d_1$ is fixed and $N_1 \rightarrow \infty$. In this asymptotic setting, it is possible to sharpen Lemma 7.3.3. Our arguments are less formal than in the rest of the paper.

Let $\mathbf{x}_0 \in \mathbb{R}^N$ be an unknown vector, and imagine we observe

$$\mathbf{y} = \mathbf{A}\mathbf{x}_0 + \mathbf{z},$$

where \mathbf{A} is a $d \times N$ matrix with i.i.d. $\mathcal{N}(0, 1/d)$ entries, and $\mathbf{z} \sim \mathcal{N}(0, \sigma^2 \mathbf{I}_d)$. Let $\hat{\mathbf{x}}$ be the solution to

$$\min \frac{1}{2} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_{\ell_2}^2 + \lambda \|\mathbf{x}\|_{\ell_1}.$$

Then setting $\delta = d/N$, the main result in [31, 33] states that almost surely,

$$\lim_{N \rightarrow \infty} \frac{1}{N} \|\hat{\mathbf{x}} - \mathbf{x}_0\|_{\ell_2}^2 = \mathbb{E}\{\left[\eta(X_0 + \tau_* Z; \alpha \tau_*) - X_0\right]^2\} = \delta(\tau_*^2 - \sigma^2),$$

where $Z \sim \mathcal{N}(0, 1)$ and the random variable X_0 has the empirical distribution of the entries of \mathbf{x}_0 . In addition, Z and X_0 are independent. We refer to [31, 33] for a precise statement.

Above, α and τ_* are solutions to

$$\lambda = \alpha\tau_* \left[1 - \frac{1}{\delta} \mathbb{E}\{\eta'(X_0 + \tau_* Z; \alpha\tau_*)\} \right] \quad (\text{D.0.1})$$

$$\tau_*^2 = \sigma^2 + \frac{1}{\delta} \mathbb{E}[\eta(X_0 + \tau_* Z; \alpha\tau_*) - X_0]^2. \quad (\text{D.0.2})$$

Here, $\eta(\mathbf{x}, \theta)$ is applying a soft-thresholding rule elementwise. For a scalar t , this rule is of the form

$$\eta(t, \theta) = \text{sgn}(t) \max(|t| - \theta, 0).$$

We apply this in the setting of Lemma 7.3.3 with $\mathbf{x}_0 = \mathbf{0}$, $X_0 = 0$. Here, $\mathbf{A} = \mathbf{U}_1^T \mathbf{Y}_{\parallel}^{(1)}$ and with abuse of notation $\mathbf{y} := \mathbf{U}_1^T \mathbf{y}_{\parallel}$. In the asymptotic regime the vector \mathbf{y} and the columns of \mathbf{A} are both random Gaussian vectors with variance of each entry equal to $1/d_1 + 1/n$. Since the LASSO solution is invariant by rescaling of the columns and we are interested in bounding its norm, we assume without loss of generality that \mathbf{y} and \mathbf{A} have $\mathcal{N}(0, 1/d)$ entries, i.e. the variance of the noise \mathbf{z} above is $1/d$. With this, the above result simplifies to

$$\lim_{N \rightarrow \infty} \frac{1}{N} \|\hat{\mathbf{x}}\|_{\ell_2}^2 = \mathbb{E}\{\eta(\tau_* Z; \alpha\tau_*)\}^2 = \delta(\tau_*^2 - \sigma^2), \quad \sigma^2 = 1/d.$$

To find α and τ_* , we solve

$$\lambda = \alpha\tau_* \left[1 - \frac{1}{\delta} \mathbb{E}\{\eta'(\tau_* Z; \alpha\tau_*)\} \right], \quad \tau_*^2 = \sigma^2 + \frac{1}{\delta} \mathbb{E}[\eta(\tau_* Z; \alpha\tau_*)]^2.$$

Now notice that

$$\mathbb{E}\{\eta'(\tau_* Z; \alpha\tau_*)\} = 2\mathbb{P}\{Z \geq \alpha\}, \quad \mathbb{E}[\eta(\tau_* Z; \alpha\tau_*)]^2 = \tau_*^2 \mathbb{E}[\eta(Z; \alpha)]^2.$$

The equations then become

$$\lambda = \alpha\tau_* \left[1 - \frac{2}{\delta} \mathbb{P}\{Z \geq \alpha\} \right], \quad \tau_*^2 = \frac{\sigma^2}{1 - \frac{1}{\delta} \mathbb{E}[\eta(Z; \alpha)]^2}.$$

Eliminating τ_* and solving for α yields

$$\lambda \sqrt{\left(1 - \frac{1}{\delta} \mathbb{E}[\eta(Z; \alpha)]^2\right)} = \alpha \sigma \left[1 - \frac{2}{\delta} \mathbb{P}\{Z \geq \alpha\}\right].$$

This one-dimensional nonlinear equation can be solved with high accuracy. Plugging in the solution in the expression for τ_* bounds the ℓ_2 norm of the solution.

Now we explain how these relationships can be used to show $\|\hat{\mathbf{x}}\|_{\ell_2} \leq 1$ for $\rho \geq \rho^*$ as $\lambda \rightarrow 0$. The argument for any $\lambda > 0$ follows along similar steps, which we avoid here. As λ tends to zero we must have

$$0 = 1 - \frac{2}{\delta} \mathbb{P}\{Z \geq \alpha\} \Rightarrow \mathbb{P}\{Z \geq \alpha\} = \frac{\delta}{2} \Rightarrow \alpha = \sqrt{2} \operatorname{erfc}^{-1}(\delta)$$

where erfc^{-1} is the inverse of $\operatorname{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-t^2} dt$. With this, we obtain

$$\|\hat{\mathbf{x}}\|_{\ell_2}^2 = N \delta (\tau_*^2 - \sigma^2) = N \delta \sigma^2 \frac{\mathbb{E}[\eta(Z; \alpha)]^2}{\delta - \mathbb{E}[\eta(Z; \alpha)]^2} = \frac{\mathbb{E}[\eta(Z; \alpha)]^2}{\delta - \mathbb{E}[\eta(Z; \alpha)]^2}.$$

Some algebraic manipulations give

$$\mathbb{E}[\eta(Z; \alpha)]^2 = (\alpha^2 + 1) \operatorname{erfc}(\alpha/\sqrt{2}) - \alpha \sqrt{\frac{2}{\pi}} e^{-\alpha^2/2} = (\alpha^2 + 1) \delta - \alpha \sqrt{\frac{2}{\pi}} e^{-\alpha^2/2},$$

where $\alpha = \sqrt{2} \operatorname{erfc}^{-1}(\delta)$. For the bound to be less than 1 it suffices to have $\mathbb{E}[\eta(Z; \alpha)]^2 \leq \delta/2$. After simplification, this is equivalent to

$$\delta = \operatorname{erfc}(\alpha/\sqrt{2}) \leq \sqrt{\frac{2}{\pi}} \frac{\alpha}{\alpha^2 + 1/2} e^{-\alpha^2/2}. \quad (\text{D.0.3})$$

The two functions on both sides of the above inequality are shown in Figure D.1. As can be seen, for $\delta \leq 0.35476$ we have the desired inequality. This is equivalent to $N_1/d_1 = \rho_1 \geq \rho^* = 2.8188$.

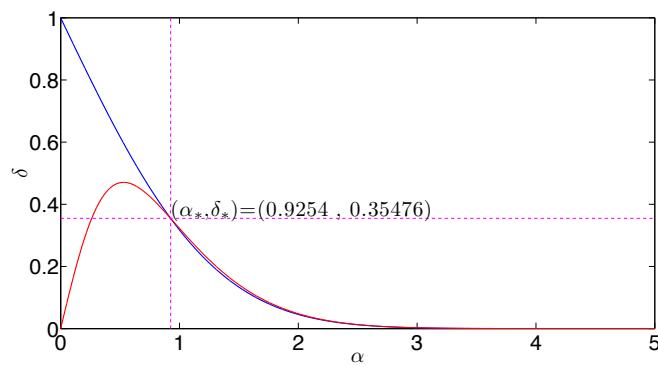


Figure D.1: Left-hand side (blue) and right-hand side (red) of (D.0.3). The two curves intersect at $(\alpha_*, \delta_*) = (0.9254, 0.35476)$.

Appendix E

Proof of auxilary lemmas for establishing the exactness of PhaseLift with CDP measurements

Set $\omega = e^{\frac{2\pi i}{n}}$ to be the n th root of unity so that

$$\mathbf{f}_k^* = [\omega^{-0(k-1)}, \omega^{-1(k-1)}, \dots, \omega^{-(n-1)(k-1)}], \quad \mathbf{f}_k = \begin{bmatrix} \omega^{0(k-1)} \\ \omega^{1(k-1)} \\ \vdots \\ \omega^{(n-1)(k-1)} \end{bmatrix}.$$

For two integers a and b we use $a \stackrel{n}{\equiv} b$ to denote congruence of a and b modulo n (n divides $a - b$).

E.1 Proof of Lemma 15.1.1

Put

$$\mathbf{Y} := \frac{1}{n} \sum_{k=1}^n |\mathbf{f}_k^* \mathbf{D}^* \mathbf{x}|^2 \mathbf{D} \mathbf{f}_k \mathbf{f}_k^* \mathbf{D}^*.$$

APPENDIX E. PROOF OF AUXILIARY LEMMAS FOR ESTABLISHING THE EXACTNESS OF

By definition,

$$\begin{aligned} |\mathbf{f}_k^* \mathbf{D}^* \mathbf{x}|^2 &= \left(\sum_{a=1}^n \bar{d}_a x_a \omega^{-(a-1)(k-1)} \right) \left(\sum_{b=1}^n d_b \bar{x}_b \omega^{(b-1)(k-1)} \right) \\ &= \sum_{a=1}^n \sum_{b=1}^n \omega^{(b-a)(k-1)} \bar{d}_a d_b x_a \bar{x}_b \end{aligned}$$

Further,

$$\begin{aligned} \mathbf{Y}_{pq} &= \frac{1}{n} \sum_{k=1}^n \sum_{a=1}^n \sum_{b=1}^n \omega^{(b-a+p-q)(k-1)} \bar{d}_a d_b d_p \bar{d}_q x_a \bar{x}_b \\ &= \sum_{a=1}^n \sum_{b=1}^n \bar{d}_a d_b d_p \bar{d}_q x_a \bar{x}_b \left(\frac{1}{n} \sum_{k=1}^n \omega^{(b-a+p-q)(k-1)} \right) \\ &= \sum_{a=1}^n \sum_{b=1}^n \bar{d}_a d_b d_p \bar{d}_q x_a \bar{x}_b \mathbb{1}_{\{a+q \equiv b+p\}}. \end{aligned}$$

Therefore,

$$\mathbb{E}[\mathbf{Y}_{pq}] = \sum_{a=1}^n \sum_{b=1}^n \mathbb{E}[\bar{d}_a d_b d_p \bar{d}_q] x_a \bar{x}_b \mathbb{1}_{\{a+q \equiv b+p\}}.$$

- Diagonal terms ($p = q$): Here, $\mathbb{E}[\bar{d}_a d_b | d_p|^2] = 0$ unless $a = b$. This gives

$$\begin{aligned} \mathbb{E}[\mathbf{Y}_{pp}] &= \sum_{a=1}^n \mathbb{E}[|d_a|^2 |d_p|^2] |x_a|^2 \\ &= \mathbb{E}[|d_p|^4] |x_p|^2 + \mathbb{E}[|d_p|^2 (\sum_{a \neq p}^n |d_a|^2 |x_a|^2)] \\ &= |x_p|^2 + \|\mathbf{x}\|_{\ell_2}^2. \end{aligned}$$

- Off-diagonal terms ($p \neq q$): Here $\mathbb{E}[\bar{d}_a d_b d_p \bar{d}_q] = 0$ unless $(a = p, b = q)$ so that

$$\mathbb{E}[\mathbf{Y}_{pq}] = (\mathbb{E}[|d|^2])^2 x_p \bar{x}_q = x_p \bar{x}_q.$$

This concludes the proof.

E.2 Proof of Lemma 15.1.2

Put

$$\mathbf{R} = \frac{1}{n} \sum_{k=1}^n (\mathbf{f}_k^* \mathbf{D}^* \mathbf{x})^2 \mathbf{D} \mathbf{f}_k \mathbf{f}_k^T \mathbf{D}.$$

By definition,

$$(\mathbf{f}_k^* \mathbf{D}^* \mathbf{x})^2 = \sum_{a=1}^n \sum_{b=1}^n \omega^{-(a+b-2)(k-1)} \bar{d}_a \bar{d}_b x_a x_b$$

and

$$\begin{aligned} \mathbf{R}_{pq} &= \frac{1}{n} \sum_{k=1}^n \sum_{a=1}^n \sum_{b=1}^n \omega^{(p+q-a-b)(k-1)} \bar{d}_a \bar{d}_b d_p d_q x_a x_b \\ &= \sum_{a=1}^n \sum_{b=1}^n \bar{d}_a \bar{d}_b d_p d_q x_a x_b \left(\frac{1}{n} \sum_{k=1}^n \omega^{(p+q-a-b)(k-1)} \right) \\ &= \sum_{a=1}^n \sum_{b=1}^n \bar{d}_a \bar{d}_b d_p d_q x_a x_b \mathbb{1}_{\{p+q \equiv a+b\}}. \end{aligned}$$

Therefore,

$$\mathbb{E}[\mathbf{R}_{pq}] = \sum_{a=1}^n \sum_{b=1}^n \mathbb{E}[\bar{d}_a \bar{d}_b d_p d_q] x_a x_b \mathbb{1}_{\{p+q \equiv a+b\}}.$$

- Diagonal terms ($p = q$): Here, $\mathbb{E}[\bar{d}_a d_b |d_p|^2] = 0$ unless $a = b = p$. This gives

$$\mathbb{E}[\mathbf{R}_{pp}] = \mathbb{E}[|d_p|^4] x_p^2 = 2x_p^2.$$

- Off-diagonal terms ($p \neq q$): Here, $\mathbb{E}[\bar{d}_a \bar{d}_b d_p d_q] = 0$ unless $(a = p, b = q)$ or $(a = q, b = p)$. This gives

$$\mathbb{E}[\mathbf{R}_{pq}] = 2\mathbb{E}[|d_p|^2 |d_q|^2] x_p x_q = 2x_p x_q.$$

This concludes the proof.

E.3 Proof of Lemma 15.1.3

Note that

$$\mathbf{Z} := \frac{1}{nL} \mathcal{A}^*(\mathbf{1}) = \frac{1}{nL} \sum_{\ell=1}^L \sum_{k=1}^n \mathbf{D}_\ell \mathbf{f}_k \mathbf{f}_k^* \mathbf{D}_\ell^* = \frac{1}{L} \sum_{\ell=1}^L \mathbf{D}_\ell \mathbf{D}_\ell^*.$$

Therefore, \mathbf{Z} is a diagonal matrix with i.i.d. diagonal entries distributed as $\frac{1}{L} \sum_{\ell=1}^L X_\ell$, where the X_ℓ are i.i.d. random variables with $\mathbb{E}[X_\ell] = \mathbb{E}[|d|^2] = 1$ and $|X_\ell| = |d|^2 \leq M^2$. The statement in the lemma then follows from Hoeffding's inequality

$$\mathbb{P}\left\{\left|\frac{1}{L} \sum_{\ell=1}^L X_\ell - 1\right| \geq t\right\} \leq 2e^{-\frac{2L}{M^2}t^2}$$

combined with the union bound.

E.4 Proof of Lemma 15.1.4

The proof is straightforward and parallels calculations in [64]. Fix a unit-normed vector \mathbf{v} , then

$$\|\mathcal{A}(\mathbf{v}\mathbf{v}^*)\|_{\ell_1} = \sum_{\ell=1}^L \sum_{k=1}^n |\mathbf{f}_k^* \mathbf{D}_\ell^* \mathbf{v}|^2 = n \sum_{\ell=1}^L \|\mathbf{D}_\ell^* \mathbf{v}\|_{\ell_2}^2 \leq nM^2 \sum_{\ell=1}^L \|\mathbf{v}\|_{\ell_2}^2 = nLM^2.$$

Consider now the eigenvalue decomposition $\mathbf{X} = \sum_{j=1}^n \lambda_j \mathbf{v}_j \mathbf{v}_j^*$ where λ_j is nonnegative since $\mathbf{X} \succeq \mathbf{0}$. Then

$$\|\mathcal{A}(\mathbf{X})\|_{\ell_1} = \sum_{j=1}^n \lambda_j \|\mathcal{A}(\mathbf{v}_j \mathbf{v}_j^*)\|_{\ell_1} \leq nLM^2 \sum_j \lambda_j = nLM^2 \text{trace}(\mathbf{X}).$$

Appendix F

Extensions of proofs of PhaseLift to higher dimensions by tensorization

Consider a two dimensional signal (image) $\mathbf{x} \in \mathbb{R}^{n_1 \times n_2}$ and assume that we have one dimensional admissible masks of the form $\mathbf{d} \in \mathbb{R}^{n_1}$ and $\mathbf{b} \in \mathbb{R}^{n_2}$.¹ We will use \mathbf{B} and \mathbf{D} to denote diagonal matrices with \mathbf{b} and \mathbf{d} on the diagonal. Then we will use 2D masks of the form $\mathbf{d}\mathbf{1}^*$ and $\mathbf{1}\mathbf{b}^*$, where $\mathbf{1}$ denotes the all one vector. Examples of such masks are depicted in Figure F.1. Using this form of masks our measurements will be of the form

$$y_{r,s,k}^{(1)} = |\mathbf{f}_{r,n_1}^* \mathbf{x} \mathbf{B}_k \mathbf{f}_{s,n_2}|^2 \quad \text{and} \quad y_{r,s,\ell}^{(2)} = |\mathbf{f}_{r,n_1}^* \mathbf{D}_\ell^* \mathbf{x} \mathbf{f}_{s,n_2}|^2$$

for $k = 1, 2, \dots, L_1$, $\ell = 1, 2, \dots, L_2$, $r = 1, 2, \dots, n_1$ and $s = 1, 2, \dots, n_2$ with $\mathbf{f}_{k,n}^*$ denoting the rows of \mathbf{F}_n the DFT matrix of \mathbb{R}^n . Now set $\mathbf{x}^{(1)} = \mathbf{F}_{n_1} \mathbf{x}$ and $\mathbf{x}^{(2)} = \mathbf{x} \mathbf{F}_{n_2}^*$. It is easy to see that we can recover the rows of $\mathbf{x}^{(1)}$ and the columns of $\mathbf{x}^{(2)}$ up to a global phase with high probability by applying the SDP feasibility problem (11.2.3) separately for each row/column (as long as $L_1 \geq c_1 \log^4 n_1$ and $L_2 \geq c_2 \log^4 n_2$ of

¹The diagonal masks we discussed before would have \mathbf{d} on the diagonal.

course). Therefore we arrive at matrices of the form

$$\begin{aligned}\tilde{\mathbf{x}}^{(1)} &= \text{diag}(e^{i\theta_1}, e^{i\theta_2}, \dots, e^{i\theta_{n_1}}) \mathbf{x}^{(1)}, \\ \tilde{\mathbf{x}}^{(2)} &= \mathbf{x}^{(2)} \text{diag}(e^{i\phi_1}, e^{i\phi_2}, \dots, e^{i\phi_{n_2}}).\end{aligned}$$

Using $\hat{\mathbf{x}} = \mathcal{F}(\mathbf{x})$ to denote the two dimensional Discrete Fourier transform of \mathbf{x} . Then we have

$$\begin{aligned}\mathbf{z}^{(1)} &:= \tilde{\mathbf{x}}^{(1)} \mathbf{F}_{n_2}^* = \text{diag}(e^{i\theta_1}, e^{i\theta_2}, \dots, e^{i\theta_{n_1}}) \hat{\mathbf{x}}, \\ \mathbf{z}^{(2)} &:= \mathbf{F}_{n_1} \tilde{\mathbf{x}}^{(2)} = \hat{\mathbf{x}} \text{diag}(e^{i\phi_1}, e^{i\phi_2}, \dots, e^{i\phi_{n_2}}).\end{aligned}\tag{F.0.1}$$

Define the bipartite graph of a matrix $\mathbf{X} = [X_{ab}] \in \mathbb{R}^{n_1 \times n_2}$ as the bipartite graph with vertices $1, 2, \dots, n_1$ (representing the rows) and $1, 2, \dots, n_2$ (representing the columns), such that there is an edge joining node a to node b if and only if $X_{ab} \neq 0$. Let a_1, a_2, \dots, a_{n_1} denote the nodes corresponding to the rows of $\hat{\mathbf{x}}$ and b_1, b_2, \dots, b_{n_2} denote the nodes corresponding to the columns of $\hat{\mathbf{x}}$. We assume that the bipartite graph of $\hat{\mathbf{x}}$ is connected. Then for any node b_r representing a column of $\hat{\mathbf{x}}$ we have a path connecting node a_1 to node b_j . That is, a path of the form $(a_{i_1}, b_{j_1}), (b_{j_1}, a_{i_2}), \dots, (a_{i_p}, b_{j_p})$ with $i_1 = 1$ and $j_p = j$. Note that when $\hat{\mathbf{x}}_{ab} \neq 0$ then we have $\mathbf{z}_{ab}^{(1)}/\mathbf{z}_{ab}^{(2)} = e^{i\theta_a}/e^{i\phi_b}$ so that if we know θ_a we can deduce ϕ_b . Applying this argument along the path from a_1 to b_j , given θ_1 we can then find ϕ_j . A similar argument holds for any node a_i . Therefore when the bipartite graph of $\hat{\mathbf{x}}$ is connected, fixing θ_1 , we can uniquely determine all parameters $\theta_2, \theta_3, \dots, \theta_{n_1}$ and $\phi_1, \phi_2, \dots, \phi_{n_2}$.² As a result it is easy to see that as long as the bipartite graph associated with $\hat{\mathbf{x}}$ is connected one can recover $\hat{\mathbf{x}}$ and therefore \mathbf{x} up to a global phase factor from the two equations in (F.0.1). We note that in the above $L = L_1 + L_2 \geq c_1 \log^4 n_1 + c_2 \log^4 n_2$ measurements are sufficient. One can also apply a similar argument to show that $(\log n_1)^4 \times (\log n_2)^4$ independent masks of the form $\mathbf{b}\mathbf{d}^*$ are sufficient without the technical assumption stated above. However, as mentioned before we should emphasize that our methods are directly applicable to the 2D case but we will not pursue this in this paper.

²We note that the complexity of solving for all the unknown parameters is linear in $n_1 + n_2$.

APPENDIX F. EXTENSIONS OF PROOFS OF PHASELIFT TO HIGHER DIMENSIONS BY TE

d_1							
d_2							
d_3							
d_4							
d_5							

b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8
b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8
b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8
b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8
b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8

Figure F.1: An example of constructing an admissible two dimensional make of size 5×8 of the form $\mathbf{d1}^*$ and $\mathbf{1b}^*$ using one dimensional admissible masks of size 5 and 8.

Appendix G

Wirtinger derivatives

Our gradient step (12.1.2) uses a notion of derivative, which can be interpreted as a Wirtinger derivative. The purpose of this section is thus to gather some results concerning Wirtinger derivatives of real valued functions over complex variables. Here and below, \mathbf{M}^T is the transpose of the matrix \mathbf{M} , and \bar{c} denotes the complex conjugate of a scalar $c \in \mathbb{C}$. Similarly, the matrix $\bar{\mathbf{M}}$ is obtained by taking complex conjugates of the elements of \mathbf{M} .

Any complex-or real-valued function

$$f(\mathbf{z}) = f(\mathbf{x}, \mathbf{y}) = u(\mathbf{x}, \mathbf{y}) + iv(\mathbf{x}, \mathbf{y})$$

of several complex variables can be written in the form $f(\mathbf{z}, \bar{\mathbf{z}})$, where f is holomorphic in $\mathbf{z} = \mathbf{x} + i\mathbf{y}$ for fixed $\bar{\mathbf{z}}$ and holomorphic in $\bar{\mathbf{z}} = \mathbf{x} - i\mathbf{y}$ for fixed \mathbf{z} . This holds as long as the real-valued functions u and v are differentiable as functions of the real variables \mathbf{x} and \mathbf{y} . As an example, consider

$$f(\mathbf{z}) = (y - |\mathbf{a}^* \mathbf{z}|^2)^2 = (y - \bar{\mathbf{z}}^T \mathbf{a} \mathbf{a}^* \mathbf{z})^2 = f(\mathbf{z}, \bar{\mathbf{z}}).$$

with $\mathbf{z}, \mathbf{a} \in \mathbb{C}^n$ and $y \in \mathbb{R}$. While $f(\mathbf{z})$ is not holomorphic in \mathbf{z} , $f(\mathbf{z}, \bar{\mathbf{z}})$ is holomorphic in \mathbf{z} for a fixed $\bar{\mathbf{z}}$, and vice versa.

This fact underlies the development of the *Wirtinger calculus*. In essence, the

conjugate coordinates

$$\begin{bmatrix} \mathbf{z} \\ \bar{\mathbf{z}} \end{bmatrix} \in \mathbb{C}^n \times \mathbb{C}^n, \quad \mathbf{z} = \mathbf{x} + i\mathbf{y} \quad \text{and} \quad \bar{\mathbf{z}} = \mathbf{x} - i\mathbf{y},$$

can serve as a formal substitute for the representation $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{2n}$. This leads to the following derivatives

$$\begin{aligned} \frac{\partial f}{\partial \mathbf{z}} &:= \frac{\partial f(\mathbf{z}, \bar{\mathbf{z}})}{\partial \mathbf{z}}|_{\bar{\mathbf{z}}=\text{constant}} = \left[\frac{\partial f}{\partial z_1}, \frac{\partial f}{\partial z_2}, \dots, \frac{\partial f}{\partial z_n} \right]_{\bar{\mathbf{z}}=\text{constant}}, \\ \frac{\partial f}{\partial \bar{\mathbf{z}}} &:= \frac{\partial f(\mathbf{z}, \bar{\mathbf{z}})}{\partial \bar{\mathbf{z}}}|_{\mathbf{z}=\text{constant}} = \left[\frac{\partial f}{\partial \bar{z}_1}, \frac{\partial f}{\partial \bar{z}_2}, \dots, \frac{\partial f}{\partial \bar{z}_n} \right]_{\mathbf{z}=\text{constant}}. \end{aligned}$$

Our definitions follow standard notation from multivariate calculus so that derivatives are row vectors and gradients are column vectors. In this new coordinate system the complex gradient is given by

$$\nabla_c f = \left[\frac{\partial f}{\partial \mathbf{z}}, \frac{\partial f}{\partial \bar{\mathbf{z}}} \right]^*.$$

Similarly, we define

$$\mathcal{H}_{zz} := \frac{\partial}{\partial \mathbf{z}} \left(\frac{\partial f}{\partial \mathbf{z}} \right)^*, \quad \mathcal{H}_{\bar{z}z} := \frac{\partial}{\partial \bar{\mathbf{z}}} \left(\frac{\partial f}{\partial \mathbf{z}} \right)^*, \quad \mathcal{H}_{z\bar{z}} := \frac{\partial}{\partial \mathbf{z}} \left(\frac{\partial f}{\partial \bar{\mathbf{z}}} \right)^*, \quad \mathcal{H}_{\bar{z}\bar{z}} := \frac{\partial}{\partial \bar{\mathbf{z}}} \left(\frac{\partial f}{\partial \bar{\mathbf{z}}} \right)^*.$$

In this coordinate system the complex Hessian is given by

$$\nabla^2 f := \begin{bmatrix} \mathcal{H}_{zz} & \mathcal{H}_{\bar{z}z} \\ \mathcal{H}_{z\bar{z}} & \mathcal{H}_{\bar{z}\bar{z}} \end{bmatrix}.$$

Given vectors \mathbf{z} and $\Delta \mathbf{z} \in \mathbb{C}^n$, we have defined the gradient and Hessian in a manner such that Taylor's approximation takes the form

$$f(\mathbf{z} + \Delta \mathbf{z}) \approx f(\mathbf{z}) + (\nabla_c f(\mathbf{z}))^* \left[\frac{\Delta \mathbf{z}}{\Delta \mathbf{z}} \right] + \frac{1}{2} \left[\frac{\Delta \mathbf{z}}{\Delta \mathbf{z}} \right]^* \nabla^2 f(\mathbf{z}) \left[\frac{\Delta \mathbf{z}}{\Delta \mathbf{z}} \right].$$

If we were to run gradient descent in this new coordinate system, the iterates would

be

$$\begin{bmatrix} \mathbf{z}_{\tau+1} \\ \bar{\mathbf{z}}_{\tau+1} \end{bmatrix} = \begin{bmatrix} \mathbf{z}_\tau \\ \bar{\mathbf{z}}_\tau \end{bmatrix} - \mu \begin{bmatrix} (\partial f / \partial \mathbf{z})^*|_{\mathbf{z}=\mathbf{z}_\tau} \\ (\partial f / \partial \bar{\mathbf{z}})^*|_{\mathbf{z}=\mathbf{z}_\tau} \end{bmatrix} \quad (\text{G.0.1})$$

Note that when f is a real-valued function (as in this paper) we have

$$\frac{\overline{\partial f}}{\partial \mathbf{z}} = \frac{\partial f}{\partial \bar{\mathbf{z}}}.$$

Therefore, the second set of updates in (G.0.1) is just the conjugate of the first. Thus, it is sufficient to keep track of the first update, namely,

$$\mathbf{z}_{\tau+1} = \mathbf{z}_\tau - \mu (\partial f / \partial \mathbf{z})^*.$$

For real valued functions of complex variables, setting

$$\nabla f(\mathbf{z}) = \left(\frac{\partial f}{\partial \mathbf{z}} \right)^*$$

gives the gradient update

$$\mathbf{z}_{\tau+1} = \mathbf{z}_\tau - \mu \nabla f(\mathbf{z}_\tau).$$

The reader may wonder why we choose to work with conjugate coordinates as there are alternatives: in particular, we could view the complex variable $\mathbf{z} = \mathbf{x} + i\mathbf{y} \in \mathbb{C}^n$ as a vector in \mathbb{R}^{2n} and just run gradient descent in the \mathbf{x}, \mathbf{y} coordinate system. The main reason why conjugate coordinates are particularly attractive is that expressions for derivatives become significantly simpler and resemble those we obtain in the real case, where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is a function of real variables.

Appendix H

Expectations and deviations

We provide here the proofs of our intermediate results for the proof of convergence of WF. Throughout this section we use $\mathbf{D} \in \mathbb{C}^{n \times n}$ to denote a diagonal random matrix with diagonal elements being i.i.d. samples from an admissible distribution d (recall the definition (11.3.1) of an admissible random variable). For ease of exposition, we shall rewrite (??) in the form

$$y_r = \left| \sum_{t=0}^{n-1} x[t] \bar{d}_\ell(t) e^{-i2\pi kt/n} \right|^2 = |\mathbf{f}_k^* \mathbf{D}_\ell^* \mathbf{x}|^2, \quad r = (\ell, k), \quad \begin{matrix} 0 \leq k \leq n-1 \\ 1 \leq \ell \leq L \end{matrix},$$

where \mathbf{f}_k^* is the k th row of the $n \times n$ DFT matrix and \mathbf{D}_ℓ is a diagonal matrix with the diagonal entries given by $d_\ell(0), d_\ell(1), \dots, d_\ell(n-1)$. In our model, the matrices \mathbf{D}_ℓ are i.i.d. copies of \mathbf{D} .

H.1 Proof of Lemma 15.3.1

The proof for admissible coded diffraction patterns follows from Lemmas 3.1 and 3.2 in [58]. For the Gaussian model, it is a consequence of the two lemmas below, whose proofs are omitted.

Lemma H.1.1 *Suppose the sequence $\{\mathbf{a}_r\}$ follows the Gaussian model. Then for*

any fixed vector $\mathbf{x} \in \mathbb{C}^n$,

$$\mathbb{E}\left(\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{x}|^2 \mathbf{a}_r \mathbf{a}_r^*\right) = \mathbf{x} \mathbf{x}^* + \|\mathbf{x}\|_{\ell_2}^2 \mathbf{I}.$$

Lemma H.1.2 Suppose the sequence $\{\mathbf{a}_r\}$ follows the Gaussian model. Then for any fixed vector $\mathbf{x} \in \mathbb{C}^n$,

$$\mathbb{E}\left(\frac{1}{m} \sum_{r=1}^m (\mathbf{a}_r^* \mathbf{x})^2 \mathbf{a}_r \mathbf{a}_r^*\right) = 2\mathbf{x} \mathbf{x}^T.$$

H.2 Proof of Lemma 15.3.2

Recall that

$$\nabla f(\mathbf{z}) = \frac{1}{m} \sum_{r=1}^m (|\langle \mathbf{a}_r, \mathbf{z} \rangle|^2 - y_r) (\mathbf{a}_r \mathbf{a}_r^*) \mathbf{z} = \frac{1}{m} \sum_{r=1}^m (|\langle \mathbf{a}_r, \mathbf{z} \rangle|^2 - |\langle \mathbf{a}_r, \mathbf{x} \rangle|^2) (\mathbf{a}_r \mathbf{a}_r^*) \mathbf{z}.$$

Thus by applying Lemma 3.1 in [58] (for the CDP model) and Lemma H.1.1 above (for the Gaussian model) we have

$$\begin{aligned} \mathbb{E}[\nabla f(\mathbf{z})] &= \frac{1}{m} \mathbb{E}\left[\sum_{r=1}^m (|\mathbf{a}_r^* \mathbf{z}|^2 \mathbf{a}_r \mathbf{a}_r^* \mathbf{z} - |\mathbf{a}_r^* \mathbf{x}|^2 \mathbf{a}_r \mathbf{a}_r^* \mathbf{z})\right] \\ &= (\mathbf{z} \mathbf{z}^* + \|\mathbf{z}\|_{\ell_2}^2 \mathbf{I}) \mathbf{z} - (\mathbf{x} \mathbf{x}^* + \mathbf{I}) \mathbf{z} \\ &= 2(\|\mathbf{z}\|_{\ell_2}^2 - 1) \mathbf{z} + (\mathbf{I} - \mathbf{x} \mathbf{x}^*) \mathbf{z}. \end{aligned}$$

H.3 Proof of Lemma 15.3.3

Suppose $\mathbf{a} \in \mathbb{C}^n \sim \mathcal{N}(0, \mathbf{I}/2) + i\mathcal{N}(0, \mathbf{I}/2)$. Since the law of \mathbf{a} is invariant by unitary transformation, we may just as well take $\mathbf{v} = \mathbf{e}_1$ and $\mathbf{u} = s_1 e^{i\phi_1} \mathbf{e}_1 + s_2 e^{i\phi_2} \mathbf{e}_2$, where s_1, s_2 are positive real numbers obeying $s_1^2 + s_2^2 = 1$. We have

$$\begin{aligned} \mathbb{E}[(\operatorname{Re}(\mathbf{u}^* \mathbf{a} \mathbf{a}^* \mathbf{v}))^2] &= \mathbb{E}[(\operatorname{Re}(s_1 e^{i\phi_1} |a_1|^2 + s_2 e^{i\phi_2} a_1 \bar{a}_2))^2] = 2s_1^2 \cos^2(\phi_1) + \frac{1}{2}s_2^2 \\ &= \frac{1}{2} + \frac{3}{2}s_1^2 \cos^2(\phi_1) - \frac{1}{2}s_1^2 \sin^2(\phi_1) = \frac{1}{2} + \frac{3}{2}(\operatorname{Re}(\mathbf{u}^* \mathbf{v}))^2 - \frac{1}{2}(\operatorname{Im}(\mathbf{u}^* \mathbf{v}))^2. \end{aligned}$$

and

$$\mathbb{E} \left[(\operatorname{Re}(\mathbf{u}^* \mathbf{a} \mathbf{a}^* \mathbf{v})) |\mathbf{a}^* \mathbf{v}|^2 \right] = \mathbb{E} \left[\left(\operatorname{Re}(s_1 e^{-i\phi_1} |a_1|^2 + s_2 e^{-i\phi_2} \bar{a}_1 a_2) \right) |a_1|^2 \right] = 2s_1 \cos(\phi_1) = 2 \operatorname{Re}(\mathbf{u}^* \mathbf{v}).$$

The identity (15.3.6) follows from standard normal moment calculations.

H.4 Proof of Lemma 15.3.4

H.4.0.7 The CDP model

Write the Hessian as

$$\mathbf{Y} := \nabla^2 f(\mathbf{x}) = \frac{1}{nL} \sum_{\ell=1}^L \sum_{k=1}^n \mathbf{W}_k(\mathbf{D}_\ell)$$

where

$$\mathbf{W}_k(\mathbf{D}) := \begin{bmatrix} \mathbf{D} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}^* \end{bmatrix} \begin{bmatrix} \mathbf{A}_k(\mathbf{D}) & \mathbf{B}_k(\mathbf{D}) \\ \overline{\mathbf{B}_k(\mathbf{D})} & \overline{\mathbf{A}_k(\mathbf{D})} \end{bmatrix} \begin{bmatrix} \mathbf{D}^* & \mathbf{0} \\ \mathbf{0} & \mathbf{D} \end{bmatrix}$$

and

$$\mathbf{A}_k(\mathbf{D}) = |\mathbf{f}_k^* \mathbf{D}^* \mathbf{x}|^2 \mathbf{f}_k \mathbf{f}_k^*, \quad \mathbf{B}_k(\mathbf{D}) = (\mathbf{f}_k^* \mathbf{D}^* \mathbf{x})^2 \mathbf{f}_k \mathbf{f}_k^T.$$

It is useful to recall that

$$\mathbb{E} \mathbf{Y} = \begin{bmatrix} \mathbf{I} + \mathbf{x} \mathbf{x}^* & 2 \mathbf{x} \mathbf{x}^T \\ 2 \bar{\mathbf{x}} \mathbf{x}^* & \mathbf{I} + \bar{\mathbf{x}} \mathbf{x}^T \end{bmatrix}.$$

Now set

$$\tilde{\mathbf{Y}} = \frac{1}{nL} \sum_{\ell=1}^L \sum_{k=1}^n \mathbf{W}_k(\mathbf{D}_\ell) \mathbb{1}_{\{|\mathbf{f}_k^* \mathbf{D}_\ell^* \mathbf{x}| \leq \sqrt{2R \log n}\}},$$

where R is a positive scalar whose value will be determined shortly, and define the events

$$\begin{aligned} E_1(R) &= \{\|\tilde{\mathbf{Y}} - \mathbb{E} \mathbf{Y}\| \leq \epsilon\}, \\ E_2(R) &= \{\tilde{\mathbf{Y}} = \mathbf{Y}\}, \\ E_3(R) &= \bigcap_{k,\ell} \{|\mathbf{f}_k^* \mathbf{D}_\ell^* \mathbf{x}| \leq \sqrt{2R \log n}\}, \\ E &= \{\|\mathbf{Y} - \mathbb{E} \mathbf{Y}\| \leq \epsilon\}. \end{aligned}$$

Note that $E_1 \cap E_2 \subset E$. Also, if $|\mathbf{f}_k^* \mathbf{D}_\ell^* \mathbf{x}| \leq \sqrt{2R \log n}$ for all pairs (k, ℓ) , then $\tilde{\mathbf{Y}} = \mathbf{Y}$ and thus $E_3 \subset E_2$. Putting all of this together gives

$$\begin{aligned} \mathbb{P}(E^c) &\leq \mathbb{P}(E_1^c \cup E_2^c) \leq \mathbb{P}(E_1^c) + \mathbb{P}(E_2^c) \leq \mathbb{P}(E_1^c) + \mathbb{P}(E_3^c) \\ &\leq \mathbb{P}(E_1^c) + \sum_{\ell=1}^{L_0} \sum_{k=1}^n \mathbb{P}(|\mathbf{f}_k^* \mathbf{D}_\ell^* \mathbf{x}| > \sqrt{2R \log n}). \end{aligned} \tag{H.4.1}$$

A slight modification to Lemma 3.9 in [58] gives $\mathbb{P}(E_1^c) \leq 1/n^3$ provided $L \geq c(R) \log^3 n$ for a sufficiently large numerical constant $c(R)$. Since Hoeffding's inequality yields $\mathbb{P}(|\mathbf{f}_k^* \mathbf{D}_\ell^* \mathbf{x}| > \sqrt{2R \log n}) \leq 2n^{-R}$, we have

$$\mathbb{P}(E^c) \leq n^{-3} + 2(nL)n^{-R}.$$

Setting $R = 4$ completes the proof.

H.4.0.8 The Gaussian model

By unitary invariance, we may take $\mathbf{x} = \mathbf{e}_1$. Letting $\mathbf{z}(1)$ be the first coordinate of a vector \mathbf{z} , to prove Lemma 15.3.4 for the Gaussian model it suffices to prove the two inequalities,

$$\left\| \frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r(1)|^2 \mathbf{a}_r \mathbf{a}_r^* - (\mathbf{I} + \mathbf{e}_1 \mathbf{e}_1^T) \right\| \leq \frac{\delta}{4} \tag{H.4.2}$$

and

$$\left\| \frac{1}{m} \sum_{r=1}^m \overline{\mathbf{a}_r(1)}^2 \mathbf{a}_r \mathbf{a}_r^T - 2\mathbf{e}_1 \mathbf{e}_1^T \right\| \leq \frac{\delta}{4}. \quad (\text{H.4.3})$$

For any $\epsilon > 0$, there is a constant $C > 0$ with the property that $m \geq C \cdot n$ implies

$$\frac{1}{m} \sum_{r=1}^m (|\mathbf{a}_r(1)|^2 - 1) \leq \epsilon, \quad \frac{1}{m} \sum_{r=1}^m (|\mathbf{a}_r(1)|^4 - 2) < \epsilon, \quad \frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r(1)|^6 < 10$$

with probability at least $1 - 3n^{-2}$; this is a consequence of Chebyshev's inequality. Moreover a union bound gives

$$\max_{1 \leq r \leq m} |\mathbf{a}_r(1)| \leq \sqrt{10 \log m}$$

with probability at least $1 - n^{-2}$. Denote by E_0 the event on which the above inequalities hold. We show that there is another event E_1 of high probability such that (H.4.2) and (H.4.3) hold on $E_0 \cap E_1$. Our proof strategy is similar to that of Theorem 39 in [232]. To prove (H.4.2), we will show that with high probability, for any $\mathbf{y} \in \mathbb{C}^n$ obeying $\|\mathbf{y}\|_{\ell_2} = 1$, we have

$$\begin{aligned} I_0(\mathbf{y}) &:= \left| \mathbf{y}^* \left(\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r(1)|^2 \mathbf{a}_r \mathbf{a}_r^* - (\mathbf{I} + \mathbf{e}_1 \mathbf{e}_1^T) \right) \mathbf{y} \right| \\ &= \left| \frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r(1)|^2 |\mathbf{a}_r^* \mathbf{y}|^2 - (1 + |\mathbf{y}(1)|^2) \right| \leq \frac{\delta}{4}. \end{aligned} \quad (\text{H.4.4})$$

For this purpose, partition \mathbf{y} in the form $\mathbf{y} = (\mathbf{y}(1), \tilde{\mathbf{y}})$ with $\tilde{\mathbf{y}} \in \mathbb{C}^{n-1}$, and decompose the inner product as

$$|\mathbf{a}_r^* \mathbf{y}|^2 = \left(|\mathbf{a}_r(1)|^2 |\mathbf{y}(1)|^2 + 2 \operatorname{Re} \left(\tilde{\mathbf{a}}_r^* \tilde{\mathbf{y}} \mathbf{a}_r(1) \overline{\mathbf{y}(1)} \right) + |\tilde{\mathbf{a}}_r^* \tilde{\mathbf{y}}|^2 \right).$$

This gives

$$I_0(\mathbf{y}) = \left| \frac{1}{m} \sum_{r=1}^m (|\mathbf{a}_r(1)|^4 - 2) |\mathbf{y}(1)|^2 + 2 \operatorname{Re} \left(\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r(1)|^2 \mathbf{a}_r(1) \overline{\mathbf{y}(1)} \tilde{\mathbf{a}}_r^* \tilde{\mathbf{y}} \right) + \frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r(1)|^2 |\tilde{\mathbf{a}}_r^* \tilde{\mathbf{y}}|^2 - \|\tilde{\mathbf{y}}\|^2 \right|$$

which follows from $|\mathbf{y}(1)|^2 + \|\tilde{\mathbf{y}}\|_{\ell_2}^2 = 1$ since \mathbf{y} has unit norm. This gives

$$\begin{aligned} I_0(\mathbf{y}) &\leq \left| \frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r(1)|^2 - 1 \right| \|\tilde{\mathbf{y}}\|_{\ell_2}^2 + \left| \frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r(1)|^4 - 2 \right| |\mathbf{y}(1)|^2 \\ &\quad + 2 \left| \frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r(1)|^2 \mathbf{a}_r(1) \overline{\mathbf{y}(1)} \tilde{\mathbf{a}}_r^* \tilde{\mathbf{y}} \right| + \left| \frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r(1)|^2 (|\tilde{\mathbf{a}}_r^* \tilde{\mathbf{y}}|^2 - \|\tilde{\mathbf{y}}\|_{\ell_2}^2) \right| \\ &\leq 2\epsilon + 2 \left| \frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r(1)|^2 \mathbf{a}_r(1) \overline{\mathbf{y}(1)} \tilde{\mathbf{a}}_r^* \tilde{\mathbf{y}} \right| + \left| \frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r(1)|^2 (|\tilde{\mathbf{a}}_r^* \tilde{\mathbf{y}}|^2 - \|\tilde{\mathbf{y}}\|_{\ell_2}^2) \right|. \end{aligned} \tag{H.4.5}$$

We now turn our attention to the last two terms of (H.4.5). For the second term, the ordinary Hoeffding's inequality (Proposition 10 in [232]) gives that for any constants δ_0 and γ , there exists a constant $C(\delta_0, \gamma)$, such that for $m \geq C(\delta_0, \gamma) \sqrt{n (\sum_{r=1}^m |\mathbf{a}_r(1)|^6)}$,

$$\left| \frac{1}{m} \sum_{r=1}^m (|\mathbf{a}_r(1)|^2 \mathbf{a}_r(1) \overline{\mathbf{y}(1)}) \tilde{\mathbf{a}}_r^* \tilde{\mathbf{y}} \right| \leq \delta_0 |\mathbf{y}(1)| \|\tilde{\mathbf{y}}\|_{\ell_2} \leq \delta_0$$

holds with probability at least $1 - 3e^{-2\gamma n}$. To control the final term, we apply the Bernstein-type inequality (Proposition 16 in [232]) to assert the following: for any positive constants δ_0 and γ , there exists a constant $C(\delta_0, \gamma)$, such that for $m \geq C(\delta_0, \gamma) (\sqrt{n (\sum_{r=1}^m |\mathbf{a}_r(1)|^4)} + n \max_{r=1}^m |\mathbf{a}_r(1)|^2)$,

$$\left| \frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r(1)|^2 (|\tilde{\mathbf{a}}_r^* \tilde{\mathbf{y}}|^2 - \|\tilde{\mathbf{y}}\|_{\ell_2}^2) \right| \leq \delta_0 \|\tilde{\mathbf{y}}\|_{\ell_2}^2 \leq \delta_0$$

holds with probability at least $1 - 2e^{-2\gamma n}$.

Therefore, for any unit norm vector \mathbf{y} ,

$$I_0(\mathbf{y}) \leq 2\epsilon + 2\delta_0 \tag{H.4.6}$$

holds with probability at least $1 - 5e^{-2\gamma n}$. By Lemma 5.4 in [232], we can bound the operator norm via an ϵ -net argument:

$$\max_{\mathbf{y} \in \mathbb{C}^n} I_0(\mathbf{y}) \leq 2 \max_{\mathbf{y} \in \mathcal{N}} I_0(\mathbf{y}) \leq 4\epsilon + 4\delta_0,$$

where \mathcal{N} is an $1/4$ -net of the unit sphere in \mathbb{C}^n . By applying the union bound and choosing appropriate δ_0 , ϵ and γ , (H.4.2) holds with probability at least $1 - 5e^{-\gamma n}$, as long as $m \geq C'(\sqrt{n \sum_{r=1}^m |\mathbf{a}_r(1)|^6} + \sqrt{n \sum_{r=1}^m |\mathbf{a}_r(1)|^4} + n \max_{1 \leq r \leq m} |\mathbf{a}_r(1)|^2)$. On E_0 this inequality follows from $m \geq C \cdot n \log n$ provided C is sufficiently large. In conclusion, (H.4.2) holds with probability at least $1 - 5e^{-\gamma n} - 4n^{-2}$.

The proof of (H.4.3) is similar. The only difference is that the random matrix is not Hermitian, so we work with

$$I_0(\mathbf{u}, \mathbf{v}) = \left| \mathbf{u}^* \left(\frac{1}{m} \sum_{r=1}^m \overline{\mathbf{a}_r(1)}^2 \mathbf{a}_r \mathbf{a}_r^T - 2\mathbf{e}_1 \mathbf{e}_1^T \right) \mathbf{v} \right|,$$

where \mathbf{u} and \mathbf{v} are unit vectors.

H.5 Proof of Corollary 15.3.5

It follows from $\|\nabla^2 f(\mathbf{x}) - \mathbb{E}[\nabla^2 f(\mathbf{x})]\| \leq \delta$ that $\nabla^2 f(\mathbf{x}) \leq \mathbb{E}[\nabla^2 f(\mathbf{x})] + \delta \mathbf{I}$. Therefore, using the fact that for any complex scalar c , $\text{Re}(c)^2 = \frac{1}{2}|c|^2 + \frac{1}{2}\text{Re}(c^2)$, we have

$$\begin{aligned} \frac{1}{m} \sum_{r=1}^m \text{Re}(\mathbf{h}^* \mathbf{a}_r \mathbf{a}_r^* \mathbf{x})^2 &= \frac{1}{4} \sum_{r=1}^m \begin{bmatrix} \mathbf{h} \\ \bar{\mathbf{h}} \end{bmatrix}^* \begin{bmatrix} |\mathbf{a}_r^* \mathbf{x}|^2 \mathbf{a}_r \mathbf{a}_r^* & (\mathbf{a}_r^* \mathbf{x})^2 \mathbf{a}_r \mathbf{a}_r^T \\ (\overline{\mathbf{a}_r^* \mathbf{x}})^2 \bar{\mathbf{a}}_r \mathbf{a}_r^* & |\mathbf{a}_r^* \mathbf{x}|^2 \bar{\mathbf{a}}_r \mathbf{a}_r^T \end{bmatrix} \begin{bmatrix} \mathbf{h} \\ \bar{\mathbf{h}} \end{bmatrix} \\ &\leq \frac{1}{4} \begin{bmatrix} \mathbf{h} \\ \bar{\mathbf{h}} \end{bmatrix}^* \left(\mathbf{I}_{2n} + \frac{3}{2} \begin{bmatrix} \mathbf{x} \\ \bar{\mathbf{x}} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \bar{\mathbf{x}} \end{bmatrix}^* - \frac{1}{2} \begin{bmatrix} \mathbf{x} \\ -\bar{\mathbf{x}} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ -\bar{\mathbf{x}} \end{bmatrix}^* \right) \begin{bmatrix} \mathbf{h} \\ \bar{\mathbf{h}} \end{bmatrix} + \frac{\delta}{4} \begin{bmatrix} \mathbf{h} \\ \bar{\mathbf{h}} \end{bmatrix}^* \begin{bmatrix} \mathbf{h} \\ \bar{\mathbf{h}} \end{bmatrix} \\ &\leq \left(\frac{1}{2} \|\mathbf{h}\|_{\ell_2}^2 + \frac{3}{2} \text{Re}(\mathbf{x}^* \mathbf{h})^2 - \frac{1}{2} \text{Im}(\mathbf{x}^* \mathbf{h})^2 \right) + \frac{\delta}{2}. \end{aligned}$$

The other inequality is established in a similar fashion.

H.6 Proof of Corollary 15.3.6

In the proof of Lemma 15.3.4, we established that with high probability,

$$\left\| \frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{x}|^2 \mathbf{a}_r \mathbf{a}_r^* - (\mathbf{x} \mathbf{x}^* + \|\mathbf{x}\|_{\ell_2}^2 \mathbf{I}) \right\| \leq \delta.$$

Therefore,

$$\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{x}|^2 \mathbf{a}_r \mathbf{a}_r^* \geq (\mathbf{x} \mathbf{x}^* + \|\mathbf{x}\|_{\ell_2}^2 \mathbf{I}) - \delta \mathbf{I}.$$

This concludes the proof of one side. The other side is similar.

H.7 Proof of Lemma 15.3.7

Note that

$$\|\nabla f(\mathbf{z}) - \mathbb{E} \nabla f(\mathbf{z})\|_{\ell_2} = \max_{\mathbf{u} \in \mathbb{C}^n, \|\mathbf{u}\|_{\ell_2}=1} \langle \mathbf{u}, \nabla f(\mathbf{z}) - \mathbb{E} \nabla f(\mathbf{z}) \rangle$$

Therefore, to establish the concentration of $\nabla f(\mathbf{z})$ around its mean we proceed by bounding $|\langle \mathbf{u}, \nabla f(\mathbf{z}) - \mathbb{E} \nabla f(\mathbf{z}) \rangle|$. From Section 15.3.2,

$$\nabla f(\mathbf{z}) = \frac{1}{m} \sum_{r=1}^m \left(|\langle \mathbf{a}_r, \mathbf{z} \rangle|^2 - y_r \right) (\mathbf{a}_r \mathbf{a}_r^*) \mathbf{z}.$$

Define $\mathbf{h} := e^{-\phi_z} \mathbf{z} - \mathbf{x}$ and $\mathbf{w} := e^{-i\phi_z} \mathbf{u}$, we have

$$\begin{aligned} \langle \mathbf{u}, \nabla f(\mathbf{z}) \rangle &= \frac{1}{m} \sum_{r=1}^m \mathbf{w}^* \left((\mathbf{a}_r^* \mathbf{x})^2 \mathbf{a}_r \mathbf{a}_r^T \right) \bar{\mathbf{h}} + \mathbf{w}^* \left(|\mathbf{a}_r^* \mathbf{x}|^2 \mathbf{a}_r \mathbf{a}_r^* \right) \mathbf{h} \\ &\quad + 2\mathbf{w}^* \left(|\mathbf{a}_r^* \mathbf{h}|^2 \mathbf{a}_r \mathbf{a}_r^* \mathbf{h} \right) \mathbf{x} + \mathbf{w}^* \left((\mathbf{a}_r^* \mathbf{h})^2 \mathbf{a}_r \mathbf{a}_r^T \right) \bar{\mathbf{x}} + \mathbf{w}^* \left(|\mathbf{a}_r^* \mathbf{h}|^2 \mathbf{a}_r \mathbf{a}_r^* \right) \mathbf{h}. \end{aligned} \quad (\text{H.7.1})$$

By Lemma 15.3.2 we also have

$$\begin{aligned} \mathbf{w}^* \mathbb{E}[\nabla f(\mathbf{z})] &= \mathbf{w}^* \left(2\mathbf{x} \mathbf{x}^T \right) \bar{\mathbf{h}} + \mathbf{w}^* \left(\mathbf{x} \mathbf{x}^* + \|\mathbf{x}\|_{\ell_2}^2 \mathbf{I} \right) \mathbf{h} \\ &\quad + 2\mathbf{w}^* \left(\mathbf{h} \mathbf{h}^* + \|\mathbf{h}\|_{\ell_2}^2 \mathbf{I} \right) \mathbf{x} + \mathbf{w}^* \left(2\mathbf{h} \bar{\mathbf{h}}^T \right) \bar{\mathbf{x}} + \mathbf{w}^* \left(\mathbf{h} \mathbf{h}^* + \|\mathbf{h}\|_{\ell_2}^2 \mathbf{I} \right) \mathbf{h}. \end{aligned} \quad (\text{H.7.2})$$

Combining (H.7.1) and (H.7.2) together with the triangular inequality and Lemma 15.3.4 give

$$\begin{aligned}
|\langle \mathbf{u}, \nabla f(\mathbf{z}) - \mathbb{E}[\nabla f(\mathbf{z})] \rangle| &\leq \left| \mathbf{w}^* \left(\frac{1}{m} \sum_{r=1}^m (\mathbf{a}_r^* \mathbf{x})^2 \mathbf{a}_r \mathbf{a}_r^T - 2\mathbf{x}\mathbf{x}^T \right) \bar{\mathbf{h}} \right| \\
&\quad + \left| \mathbf{w}^* \left(\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{x}|^2 \mathbf{a}_r \mathbf{a}_r^* - (\mathbf{x}\mathbf{x}^* + \|\mathbf{x}\|_{\ell_2}^2 \mathbf{I}) \right) \mathbf{h} \right| \\
&\quad + 2 \left| \mathbf{w}^* \left(\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^2 \mathbf{a}_r \mathbf{a}_r^* - (\mathbf{h}\mathbf{h}^* + \|\mathbf{h}\|_{\ell_2}^2 \mathbf{I}) \right) \mathbf{x} \right| \\
&\quad + \left| \mathbf{w}^* \left(\frac{1}{m} \sum_{r=1}^m (\mathbf{a}_r^* \mathbf{h})^2 \mathbf{a}_r \mathbf{a}_r^T - 2\mathbf{h}\bar{\mathbf{h}}^T \right) \bar{\mathbf{x}} \right| \\
&\quad + \left| \mathbf{w}^* \left(\frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^2 \mathbf{a}_r \mathbf{a}_r^* - (\mathbf{h}\mathbf{h}^* + \|\mathbf{h}\|_{\ell_2}^2 \mathbf{I}) \right) \mathbf{h} \right| \\
&\leq \left\| \frac{1}{m} \sum_{r=1}^m (\mathbf{a}_r^* \mathbf{x})^2 \mathbf{a}_r \mathbf{a}_r^T - 2\mathbf{x}\mathbf{x}^T \right\| \|\mathbf{h}\|_{\ell_2} + \left\| \frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{x}|^2 \mathbf{a}_r \mathbf{a}_r^* - (\mathbf{x}\mathbf{x}^* + \|\mathbf{x}\|_{\ell_2}^2 \mathbf{I}) \right\| \\
&\quad + 2 \left\| \frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^2 \mathbf{a}_r \mathbf{a}_r^* - (\mathbf{h}\mathbf{h}^* + \|\mathbf{h}\|_{\ell_2}^2 \mathbf{I}) \right\| + \left\| \frac{1}{m} \sum_{r=1}^m (\mathbf{a}_r^* \mathbf{h})^2 \mathbf{a}_r \mathbf{a}_r^T - 2\mathbf{h}\bar{\mathbf{h}}^T \right\| \\
&\quad + \left\| \frac{1}{m} \sum_{r=1}^m |\mathbf{a}_r^* \mathbf{h}|^2 \mathbf{a}_r \mathbf{a}_r^* - (\mathbf{h}\mathbf{h}^* + \|\mathbf{h}\|_{\ell_2}^2 \mathbf{I}) \right\| \|\mathbf{h}\|_{\ell_2} \\
&\leq 3\delta \|\mathbf{h}\|_{\ell_2} (1 + \|\mathbf{h}\|_{\ell_2}) \\
&\leq \frac{9}{2}\delta \|\mathbf{h}\|_{\ell_2}.
\end{aligned}$$

H.8 Proof of Lemma 15.3.8

The result for the CDP model follows from Lemma 3.3 in [58]. For the Gaussian model, it is a consequence of standard results, e.g. Theorem 5.39 in [232], concerning the deviation of the sample covariance matrix from its mean.

Appendix I

The Power Method

We use the power method (Algorithm 9) with a random initialization to compute the first eigenvector of $\mathbf{Y} = \mathbf{A} \operatorname{diag}\{\mathbf{y}\} \mathbf{A}^*$. Since, each iteration of the power method asks to compute the matrix-vector product

$$\mathbf{Y}\mathbf{z} = \mathbf{A} \operatorname{diag}\{\mathbf{y}\} \mathbf{A}^* \mathbf{z},$$

we simply need to apply \mathbf{A} and \mathbf{A}^* to an arbitrary vector. In the Gaussian model, this costs $2mn$ multiplications while in the CDP model the cost is that of $2L$ n -point FFTs. We now turn our attention to the number of iterations required to achieve a sufficiently accurate initialization.

Algorithm 9 Power Method

Input: Matrix \mathbf{Y}

\mathbf{v}_0 is a random vector on the unit sphere of \mathbb{C}^n

for $\tau = 1$ **to** T **do**

$$\mathbf{v}_\tau = \frac{\mathbf{Y}\mathbf{v}_{\tau-1}}{\|\mathbf{Y}\mathbf{v}_{\tau-1}\|_{\ell_2}}$$

end for

Output: $\tilde{\mathbf{z}}_0 = \mathbf{v}_T$

Standard results from numerical linear algebra show that after k iterations of the power method, the accuracy of the eigenvector is $\mathcal{O}(\tan \theta_0 (\lambda_2/\lambda_1)^k)$, where λ_1 and λ_2 are the top two eigenvalues of the positive semidefinite matrix \mathbf{Y} , and θ_0 is the angle

between the initial guess and the top eigenvector. Hence, we would need on the order of $\log(n/\epsilon) / \log(\lambda_1/\lambda_2)$ for ϵ accuracy. Under the stated assumptions, Lemma 15.3.4 bounds below the eigenvalue gap by a numerical constant so that we can see that few iterations of the power method would yield accurate estimates.

Bibliography

- [1] www.stanford.edu/~mahdisol/RSC.
- [2] D. G. Mixon blog: Saving phase: Injectivity and stability for phase retrieval.
- [3] D. G. Mixon blog: AIM workshop: frame theory intersects geometry.
- [4] J. P. Abrahams and A. G. W. Leslie. Methods used in the structure determination of bovine mitochondrial f1 atpase. *Acta Crystallographica Section D: Biological Crystallography*, 52(1):30–42, 1996.
- [5] A. Agarwal, S. Negahban, and M. J. Wainwright. Fast global convergence of gradient methods for high-dimensional statistical recovery. *The Annals of Statistics*, 40(5):2452–2482, 2012.
- [6] P. K. Agarwal and N. H. Mustafa. k -means projective clustering. In *Proceedings of the twenty-third ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 155–165, 2004.
- [7] S. Agarwal, J. Lim, L. Zelnik-Manor, P. Perona, D. Kriegman, and S. Belongie. Beyond pairwise clustering. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 838–845. IEEE, 2005.
- [8] A. Ahmed, B. Recht, and J. Romberg. Blind deconvolution using convex programming. *arXiv preprint arXiv:1211.5608*, 2012.

- [9] A. Aldroubi and A. Sekmen. Nearness to local subspace algorithm for subspace and motion segmentation. *Signal Processing Letters, IEEE*, 19(10):704–707, 2012.
- [10] A. Aldroubi and K. Zaringhalam. Nonlinear least squares in \mathbb{R}^n . *Acta applicandae mathematicae*, 107(1-3):325–337, 2009.
- [11] B. Alexeev, A. S. Bandeira, M. Fickus, and D. G. Mixon. Phase retrieval with polarization. *arXiv preprint arXiv:1210.7752*, 2012.
- [12] D. Alonso-Gutiérrez. On the isotropy constant of random convex sets. *Proceedings of the American Mathematical Society*, 136(9):3293–3300, 2008.
- [13] D. Amelunxen, M. Lotz, M. B. McCoy, and J. A. Tropp. Living on the edge: Phase transitions in convex programs with random data. *Inform. Inference*, 2014.
- [14] E. Arias-Castro. Clustering based on pairwise distances when the data is of mixed dimensions. *IEEE Transactions on Information Theory*, 57(3):1692–1706, 2011.
- [15] E. Arias-Castro, G. Chen, and G. Lerman. Spectral clustering based on local linear approximations. *Electronic Journal of Statistics*, 5:1537–1587, 2011.
- [16] L. Bako. Identification of switched linear systems via sparse optimization. *Automatica*, 2011.
- [17] R. Balan. A nonlinear reconstruction algorithm from absolute value of frame coefficients for low redundancy frames. In *International Conference on Sampling Theory and Applications (SAMPTA)*, 2009.
- [18] R. Balan. On signal reconstruction from its spectrogram. In *44th Annual Conference on Information Sciences and Systems (CISS)*, pages 1–4, 2010.
- [19] R. Balan. Stability of phase retrievable frames. *arXiv preprint arXiv:1308.5465*, 2013.

- [20] R. Balan, B. G. Bodmann, P. G. Casazza, and D. Edidin. Painless reconstruction from magnitudes of frame coefficients. *Journal of Fourier Analysis and Applications*, 15(4):488–501, 2009.
- [21] R. Balan, P. Casazza, and D. Edidin. On signal reconstruction without phase. *Applied and Computational Harmonic Analysis*, 20(3):345–356, 2006.
- [22] R. Balan, P. Casazza, and D. Edidin. Equivalence of reconstruction from the absolute value of the frame coefficients to a sparse representation problem. *Signal Processing Letters, IEEE*, 14(5):341–343, 2007.
- [23] R. Balan and Y. Wang. Invertibility and robustness of phaseless reconstruction. *arXiv preprint arXiv:1308.4718*, 2013.
- [24] K. Ball and A. Pajor. Convex bodies with few faces. *Proceedings of the American Mathematical Society*, pages 225–231, 1990.
- [25] L. Balzano and S. J. Wright. Local convergence of an algorithm for subspace identification from partial data. *arXiv preprint arXiv:1306.3391*, 2013.
- [26] A. S. Bandeira, J. Cahill, D. G. Mixon, and A. A. Nelson. Saving phase: Injectivity and stability for phase retrieval. *arXiv preprint arXiv:1302.4618*, 2013.
- [27] A. S. Bandeira, Y. Chen, and D. G. Mixon. Phase retrieval from power spectra of masked signals. *arXiv preprint arXiv:1303.4458*, 2013.
- [28] Y. Barbotin and M. Vetterli. Fast and robust parametric estimation of jointly sparse channels. *Emerging and Selected Topics in Circuits and Systems, IEEE Journal on*, 2(3):402–412, 2012.
- [29] R. Bates. Fourier phase problems are uniquely solvable in more than one dimension. 1. underlying theory. *Optik*, 61(3):247–262, 1982.
- [30] H. H. Bauschke, P. L. Combettes, and D. R. Luke. Hybrid projection–reflection method for phase retrieval. *JOSA A*, 20(6):1025–1034, 2003.

- [31] M. Bayati and A. Montanari. The dynamics of message passing on dense graphs, with applications to compressed sensing. *IEEE Transactions on Information Theory*, 57(2):764–785, 2011.
- [32] M. Bayati and A. Montanari. The dynamics of message passing on dense graphs, with applications to compressed sensing. *Information Theory, IEEE Transactions on*, 57(2):764–785, 2011.
- [33] M. Bayati and A. Montanari. The LASSO risk for Gaussian matrices. *IEEE Transactions on Information Theory*, 58(4):1997 –2017, April 2012.
- [34] M. Bayati and A. Montanari. The LASSO risk for Gaussian matrices. *Information Theory, IEEE Transactions on*, 58(4):1997–2017, 2012.
- [35] S.R. Becker, E.J. Candès, and M.C. Grant. Templates for convex cone problems with applications to sparse signal recovery. *Mathematical Programming Computation*, pages 1–54, 2011.
- [36] V. Bentkus. An inequality for tail probabilities of martingales with differences bounded from one side. *Journal of Theoretical Probability*, 16(1):161–173, 2003.
- [37] A. Beurling. Sur les intégrales de Fourier absolument convergentes et leur application à une transformation fonctionnelle. In *Ninth Scandinavian Mathematical Congress*, pages 345–366, 1938.
- [38] A. Beurling. Interpolation for an interval in r^1 . *The collected Works of Arne Beurling*, 2, 1989.
- [39] A. Beurling and L. Carleson. *The collected works of Arne Beurling: Complex analysis*, volume 1. Birkhauser, 1989.
- [40] B. N. Bhaskar and B. Recht. Atomic norm denoising with applications to line spectral estimation. In *Communication, Control, and Computing (Allerton), 2011 49th Annual Allerton Conference on*, pages 261–268. IEEE, 2011.

- [41] P. J. Bickel, Y. Ritov, and A. B. Tsybakov. Simultaneous analysis of lasso and dantzig selector. *The Annals of Statistics*, 37(4):1705–1732, 2009.
- [42] A. Björner. Subspace arrangements. In *First European Congress of Mathematics*, pages 321–370. Springer, 1994.
- [43] A. Björner, I. Peeva, and J. Sidman. Subspace arrangements defined by products of linear forms. *Journal of the London Mathematical Society*, 71(2):273–288, 2005.
- [44] B. G. Bodmann and N. Hammen. Stable phase retrieval with low-redundancy frames. *Advances in Computational Mathematics*, pages 1–15, 2013.
- [45] T. E. Boult and Lisa G. Brown. Factorization-based segmentation of motions. In *Visual Motion, 1991., Proceedings of the IEEE Workshop on*, pages 179–186. IEEE, 1991.
- [46] T. E. Boult and L. Gottesfeld Brown. Factorization-based segmentation of motions. In *Proceedings of the IEEE Workshop on Visual Motion*, pages 179–186, 1991.
- [47] P. S. Bradley and O. L. Mangasarian. k -plane clustering. *Journal of Global Optimization*, 16(1):23–32, 2000.
- [48] Y. M. Bruck and L. G. Sodin. On the ambiguity of the image reconstruction problem. *Optics Communications*, 30(3):304–308, 1979.
- [49] O. Bunk, A. Diaz, F. Pfeiffer, C. David, B. Schmitt, D. K. Satapathy, and J. F. Veen. Diffractive imaging for periodic samples: retrieving one-dimensional concentration profiles across microfluidic channels. *Acta Crystallographica Section A: Foundations of Crystallography*, 63(4):306–314, 2007.
- [50] J. Cahill, P. G. Casazza, J. Peterson, and L. Woodland. Phase retrieval by projections.

- [51] E. J. Candes, Y. C. Eldar, D. Needell, and P. Randall. Compressed sensing with coherent and redundant dictionaries. *Applied and Computational Harmonic Analysis*, 31(1):59–73, 2011.
- [52] E. J. Candes, Y. C. Eldar, T. Strohmer, and V. Voroninski. Phase retrieval via matrix completion. *SIAM Journal on Imaging Sciences*, 6(1):199–225, 2013.
- [53] E. J. Candes and C. Fernandez-Granda. Super-resolution from noisy data. *Journal of Fourier Analysis and Applications*, 19(6):1229–1254, 2013.
- [54] E. J. Candes and C. Fernandez-Granda. Towards a mathematical theory of super-resolution. *Communications on Pure and Applied Mathematics*, 2013.
- [55] E. J. Candes and X. Li. Solving quadratic equations via phaselift when there are about as many equations as unknowns. *Foundations of Computational Mathematics*, pages 1–10, 2012.
- [56] E. J. Candes, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):11, 2011.
- [57] E. J. Candes, X. Li, and M. Soltanolkotabi. Phase retrieval via wirtinger flow: Theory and algorithms. *arXiv preprint arXiv:1407.1065*, 2014.
- [58] E. J. Candès, X. Li, and M. Soltanolkotabi. Phase retrieval from coded diffraction patterns. *arXiv:1310.3240*, Preprint 2013.
- [59] E. J. Candès, L. Mackey, and M. Soltanolkotabi. From subspace clustering to full-rank matrix completion. *Preprint*, 2013.
- [60] E. J. Candès and Y. Plan. Near-ideal model selection by ℓ_1 minimization. *The Annals of Statistics*, 37(5A):2145–2177, 2009.
- [61] E. J. Candès and Y. Plan. A probabilistic and RIPless theory of compressed sensing. *IEEE Transactions on Information Theory*, (99):1–1, 2010.
- [62] E. J. Candes and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717–772, 2009.

- [63] E. J. Candes, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *Information Theory, IEEE Transactions on*, 52(2):489–509, 2006.
- [64] E. J. Candes, T. Strohmer, and V. Voroninski. Phaselift: Exact and stable signal recovery from magnitude measurements via convex programming. *Communications on Pure and Applied Mathematics*, 2012.
- [65] E. J. Candes and T. Tao. Decoding by linear programming. *IEEE Transactions on Information Theory*, 51(12):4203–4215, 2005.
- [66] E. J. Candès and T. Tao. The Dantzig Selector: Statistical estimation when p is much larger than n . *The Annals of Statistics*, 35(6):2313–2351, 2007.
- [67] C. Caratheodory. Über den variabilitätsbereich der koeffizienten von potenzreihen, die gegebene werte nicht annehmen. *Mathematische Annalen*, 64(1):95–115, 1907.
- [68] L. Carleson and A. Beurling. *The collected works of Arne Beurling:[in 2 Bden]. 2. Harmonic analysis*. Birkhäuser, 1989.
- [69] A. Chai, M. Moscoso, and G. Papanicolaou. Array imaging using intensity-only measurements. *Inverse Problems*, 27(1):015005, 2011.
- [70] H. N. Chapman, A. Barty, M. J. Bogan, S. Boutet, M. Frank, S. P. Hau-Riege, S. Marchesini, B. W. Woods, S. Bajt, W. H. Benner, et al. Femtosecond diffractive imaging with a soft-x-ray free-electron laser. *Nature Physics*, 2(12):839–843, 2006.
- [71] H. N. Chapman, A. Barty, S. Marchesini, A. Noy, S. P. Hau-Riege, C. Cui, M. R. Howells, R. Rosen, H. He, J. Spence, et al. High-resolution ab initio three-dimensional x-ray diffraction microscopy. *JOSA A*, 23(5):1179–1200, 2006.
- [72] G. Chen and G. Lerman. Foundations of a multi-way spectral clustering framework for hybrid linear modeling. *Foundations of Computational Mathematics*, 9(5):517–558, 2009.

- [73] G. Chen and G. Lerman. Spectral Curvature Clustering (SCC). *International Journal of Computer Vision*, 81(3):317–330, 2009.
- [74] T. Chen, W. Yin, X. S. Zhou, D. Comaniciu, and T. S. Huang. Total variation models for variable lighting face recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(9):1519–1524, 2006.
- [75] Y. Chen, N.M. Nasrabadi, and T.D. Tran. Hyperspectral image classification using dictionary-based sparse representation. *IEEE Transactions on Geoscience and Remote Sensing*, (99):1–13, 2011.
- [76] T. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. Zheng. NUS-WIDE: a real-world web image database from national university of singapore. In *Proceedings of the ACM international conference on image and video retrieval*, page 48. ACM, 2009.
- [77] A. Conca, D. Edidin, M. Hering, and C. Vinzant. An algebraic characterization of injectivity in phase retrieval. *arXiv preprint arXiv:1312.0158*, 2013.
- [78] J. V. Corbett. The Pauli problem, state reconstruction and quantum-real numbers. *Reports on Mathematical Physics*, 57(1):53–68, 2006.
- [79] J. P. Costeira and T. Kanade. A multibody factorization method for independently moving objects. *International Journal of Computer Vision*, 29(3):159–179, 1998.
- [80] T. M. Cover and J. A. Thomas. *Elements of information theory*. John Wiley & Sons, 2012.
- [81] J. C. Dainty and J. R. Fienup. Phase retrieval and image reconstruction for astronomy. *Image Recovery: Theory and Application*, ed. by H. Stark, Academic Press, San Diego, pages 231–275, 1987.
- [82] L. Demanet and P. Hand. Stable optimizationless recovery from phaseless linear measurements. *arXiv preprint arXiv:1208.1803*, 2012.

- [83] L. Demanet and V. Jugnon. Convex recovery from interferometric measurements. *arXiv preprint arXiv:1307.6864*, 2013.
- [84] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 1–38, 1977.
- [85] H. Derksen. Hilbert series of subspace arrangements. *Journal of pure and applied algebra*, 209(1):91–98, 2007.
- [86] D. L. Donoho. Superresolution via sparsity constraints. *SIAM Journal on Mathematical Analysis*, 23(5):1309–1331, 1992.
- [87] D. L. Donoho. Compressed sensing. *Information Theory, IEEE Transactions on*, 52(4):1289–1306, 2006.
- [88] D. L. Donoho, M. Gavish, and A. Montanari. The phase transition of matrix recovery from gaussian measurements matches the minimax mse of matrix denoising. *Proceedings of the National Academy of Sciences*, 110(21):8405–8410, 2013.
- [89] D. L. Donoho, I. M. Johnstone, and A. Montanari. Accurate prediction of phase transitions in compressed sensing via a connection to minimax denoising. *arXiv preprint arXiv:1111.1041*, 2011.
- [90] D. L. Donoho and B. F. Logan. Signal recovery and the large sieve. *SIAM Journal on Applied Mathematics*, 52(2):577–591, 1992.
- [91] D. L. Donoho, A. Maleki, and A. Montanari. Message-passing algorithms for compressed sensing. *Proceedings of the National Academy of Sciences*, 106(45):18914–18919, 2009.
- [92] D. L. Donoho and J. Tanner. Sparse nonnegative solution of underdetermined linear equations by linear programming. *Proceedings of the National Academy of Sciences of the United States of America*, 102(27):9446–9451, 2005.

- [93] S. Eisebitt, M. Lorgen, W. Eberhardt, J. Luning, S. Andrews, and J. Stohr. Scalable approach for lensless imaging at x-ray wavelengths. *Applied physics letters*, 84(17):3373–3375, 2004.
- [94] Y. C. Eldar and S. Mendelson. Phase retrieval: Stability and recovery guarantees. *arXiv preprint arXiv:1211.0872*, 2012.
- [95] E. Elhamifar and R. Vidal. Clustering disjoint subspaces via sparse representation. In *IEEE International Conference on Acoustics Speech and Signal Processing, ICASSP 2010.*, pages 1926–1929. IEEE.
- [96] E. Elhamifar and R. Vidal. Sparse subspace clustering. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009.*, pages 2790–2797.
- [97] E. Elhamifar and R. Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *to appear in IEEE Transactions on Pattern Analysis and Machine Intelligence, arXiv preprint arXiv:1203.1005*, 2012.
- [98] E. Elhamifar and R. Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(11):2765–2781, 2013.
- [99] V. Elser. Phase retrieval by iterated projections. *JOSA A*, 20(1):40–55, 2003.
- [100] B. Eriksson, L. Balzano, and R. Nowak. High-rank matrix completion and subspace clustering with missing data. *Arxiv preprint arXiv:1112.5629*, 2011.
- [101] P. Favaro, R. Vidal, and A. Ravichandran. A closed form solution to robust subspace estimation and clustering. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1801–1807. IEEE, 2011.
- [102] M. A. Fiddy and U. Shahid. Legacies of the Gerchberg-Saxton algorithm. *Ultramicroscopy*, 134:48–54, 2013.
- [103] J. R. Fienup. Reconstruction of an object from the modulus of its Fourier transform. *Optics letters*, 3(1):27–29, 1978.

- [104] J. R. Fienup. Fine resolution imaging of space objects. *Final Scientific Report, 1 Oct. 1979-31 Oct. 1981 Environmental Research Inst. of Michigan, Ann Arbor. Radar and Optics Div.*, 1, 1982.
- [105] J. R. Fienup. Phase retrieval algorithms: a comparison. *Applied optics*, 21(15):2758–2769, 1982.
- [106] J. R. Fienup. Comments on “The reconstruction of a multidimensional sequence from the phase or magnitude of its Fourier transform”. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 31(3):738–739, 1983.
- [107] J. Finkelstein. Pure-state informationally complete and really complete measurements. *Physical Review A*, 70(5):052107, 2004.
- [108] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.
- [109] F. Fogel, I. Waldspurger, and A. d’Aspremont. Phase retrieval for imaging problems. *arXiv preprint arXiv:1304.7735*, 2013.
- [110] J. J. Fuchs. Sparsity and uniqueness for some specific under-determined linear systems. In *Acoustics, Speech, and Signal Processing, 2005. Proceedings.(ICASSP’05). IEEE International Conference on*, volume 5, pages v–729. IEEE, 2005.
- [111] K. J. Gaffney and H. N. Chapman. Imaging atomic structure and dynamics with ultrafast x-ray scattering. *Science*, 316(5830):1444–1448, 2007.
- [112] D. Gale. Neighborly and cyclic polytopes. In *Proc. Sympos. Pure Math*, volume 7, pages 225–232, 1963.
- [113] C. W. Gear. Multibody grouping from motion images. *International Journal of Computer Vision*, 29(2):133–150, 1998.

- [114] R. W. Gerchberg. A practical algorithm for the determination of phase from image and diffraction plane pictures. *Optik*, 35:237, 1972.
- [115] M. X. Goemans and D. P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM (JACM)*, 42(6):1115–1145, 1995.
- [116] A. Goh and R. Vidal. Segmenting motions of different types by unsupervised manifold clustering. In *IEEE International Conference on Computer Vision and Pattern Recognition, CVPR 2007.*, pages 1–6. IEEE.
- [117] J. W. Goodman and S. C. Gustafson. Introduction to fourier optics. *Optical Engineering*, 35(5):1513–1513, 1996.
- [118] V. M. Govindu. A tensor decomposition for geometric grouping and segmentation. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 1150–1157. IEEE, 2005.
- [119] D. Gross. Recovering low-rank matrices from few coefficients in any basis. *Information Theory, IEEE Transactions on*, 57(3):1548–1566, 2011.
- [120] D. Gross, F. Krahmer, and R. Kueng. A partial derandomization of phaselift using spherical designs. *arXiv preprint arXiv:1310.2267*, 2013.
- [121] D. Gross, F. Krahmer, and R. Kueng. Improved recovery guarantees for phase retrieval from coded diffraction patterns. *arXiv preprint arXiv:1402.6286*, 2014.
- [122] M. Hardt. On the provable convergence of alternating minimization for matrix completion. *arXiv preprint arXiv:1312.0925*, 2013.
- [123] R. W. Harrison. Phase problem in crystallography. *JOSA A*, 10(5):1046–1055, 1993.
- [124] T. Hastie and P. Y. Simard. Metrics and models for handwritten character recognition. *Statistical Science*, pages 54–65, 1998.

- [125] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning*, volume 2. Springer, 2009.
- [126] M. H. Hayes. The reconstruction of a multidimensional sequence from the phase or magnitude of its Fourier transform. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 30(2):140–154, 1982.
- [127] R. Heckel and H. Bolcskei. Robust subspace clustering via thresholding. *arXiv preprint arXiv:1307.4891*, 2013.
- [128] R. Heckel and H. Bolcskei. Subspace clustering via thresholding and spectral clustering. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 3263–3267. IEEE, 2013.
- [129] T. Heinosaari, L. Mazzarella, and M. M. Wolf. Quantum tomography under prior information. *Communications in Mathematical Physics*, 318(2):355–374, 2013.
- [130] M. Hirsch, S. Harmeling, S. Sra, and B Scholkopf. Online multi-frame blind deconvolution with super-resolution and saturation correction. *Astronomy and Astrophysics-Les Ulis*, 531:1217, 2011.
- [131] J. Ho, M. H. Yang, J. Lim, K. C. Lee, and D. Kriegman. Clustering appearances of objects under varying illumination conditions. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 1, pages I–11. IEEE, 2003.
- [132] K. Huang, Y. Ma, and R. Vidal. Minimum effective dimension for mixtures of subspaces: A robust gPCA algorithm and its applications. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–631. IEEE, 2004.
- [133] N. Ichimura. Motion segmentation based on factorization method and discriminant criterion. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 1, pages 600–605. IEEE, 1999.

- [134] K. Jaganathan, S. Oymak, and B. Hassibi. On robust phase retrieval for sparse signals. In *50th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 794–799, 2012.
- [135] P. Jain, P. Netrapalli, and S. Sanghavi. Low-rank matrix completion using alternating minimization. In *Proceedings of the 45th annual ACM symposium on Symposium on theory of computing*, pages 665–674. ACM, 2013.
- [136] I. M. Johnstone. Multivariate analysis and jacobi ensembles: Largest eigenvalue, tracy–widom limits and rates of convergence. *Annals of statistics*, 36(6):2638, 2008.
- [137] K. Kanatani. Geometric information criterion for model selection. *International Journal of Computer Vision*, 26(3):171–189, 1998.
- [138] K. Kanatani. Motion segmentation by subspace separation and model selection. *image*, 1:1, 2001.
- [139] K. Kanatani and C. Matsunaga. Estimating the number of independent motions for multibody motion segmentation. In *Asian Conference on Computer Vision*, pages 7–12. Citeseer, 2002.
- [140] R. H. Keshavan. *Efficient algorithms for collaborative filtering*. PhD thesis, Stanford University, 2012.
- [141] R. H. Keshavan, A. Montanari, and S. Oh. Matrix completion from a few entries. *Information Theory, IEEE Transactions on*, 56(6):2980–2998, 2010.
- [142] R. H. Keshavan, A. Montanari, and S. Oh. Matrix completion from noisy entries. *The Journal of Machine Learning Research*, 99:2057–2078, 2010.
- [143] B. Klartag and R. Vershynin. Small ball probability and dvoretzkys theorem. *Israel Journal of Mathematics*, 157(1):193–207, 2007.
- [144] Y. P. C. Kotropoulos and G. R. Arce. ℓ_1 -graph based music structure analysis. In *International Society for Music Information Retrieval Conference, ISMIR 2011*.

- [145] F. Lauer and C. Schnorr. Spectral clustering of linear subspaces for motion segmentation. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 678–685. IEEE, 2009.
- [146] K. Lee, Y. Wu, and Y. Bresler. Near optimal compressed sensing of sparse rank-one matrices via sparse power factorization. *arXiv preprint arXiv:1312.0525*, 2013.
- [147] G. Lerman and T. Zhang. Robust recovery of multiple subspaces by geometric lp minimization. *The Annals of Statistics*, 39(5):2686–2715, 2011.
- [148] X. Li and V. Voroninski. Sparse signal recovery from quadratic measurements via convex programming. *arXiv preprint arXiv:1209.4785*, 2012.
- [149] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma. Robust recovery of subspace structures by low-rank representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):171–184, Jan. 2013.
- [150] G. Liu, H. Xu, and S. Yan. Exact subspace segmentation and outlier detection by low-rank representation. *arXiv preprint arXiv:1109.1646*, 2011.
- [151] E. G. Loewen and E. Popov. *Diffraction gratings and applications*. CRC Press, 1997.
- [152] B. F. Logan. *Properties of high-pass signals*. PhD thesis, Columbia university., 1965.
- [153] B. F. Logan. Information in the zero crossings of bandpass signals. *Bell System Technical Journal*, 56(4):487–510, 1977.
- [154] B. F Logan and L. A. Shepp. Optimal reconstruction of a function from its projections. *Duke Math. J*, 42(4):645–659, 1975.
- [155] P. Loh and M. J. Wainwright. Regularized M-estimators with nonconvexity: Statistical and algorithmic theory for local optima. In *Advances in Neural Information Processing Systems*, pages 476–484, 2013.

- [156] P.L. Loh and M.J. Wainwright. High-dimensional regression with noisy and missing data: Provable guarantees with nonconvexity. *The Annals of Statistics*, 40(3):1637–1664, 2012.
- [157] L. Lu and R. Vidal. Combined central and subspace clustering for computer vision applications. In *Proceedings of the 23rd international conference on Machine learning*, pages 593–600. ACM, 2006.
- [158] Y. Ma, H. Derksen, W. Hong, and J. Wright. Segmentation of multivariate mixed data via lossy data coding and compression. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(9):1546–1562, 2007.
- [159] Y. Ma and R. Vidal. Identification of deterministic switched arx systems via identification of algebraic varieties. *Hybrid Systems: Computation and Control*, pages 449–465, 2005.
- [160] Y. Ma, A.Y. Yang, H. Derksen, and R. Fossum. Estimation of subspace arrangements with applications in modeling and segmenting mixed data. *SIAM review*, 50(3):413–458, 2008.
- [161] S. Marchesini. Invited article: A unified evaluation of iterative projection algorithms for phase retrieval. *Review of Scientific Instruments*, 78(1):011301, 2007.
- [162] S. Marchesini. Phase retrieval and saddle-point optimization. *JOSA A*, 24(10):3289–3296, 2007.
- [163] S. Marchesini, H. He, H. N. Chapman, S. P. Hau-Riege, A. Noy, M. R. Howells, U. Weierstall, and J. Spence. X-ray image reconstruction from a diffraction pattern alone. *Physical Review B*, 68(14):140101, 2003.
- [164] S. Marchesini, Y. C. Tu, and H. Wu. Alternating projection, ptychographic imaging and phase synchronization. *arXiv preprint arXiv:1402.0550*, 2014.

- [165] M. B. McCoy and J. A. Tropp. Sharp recovery bounds for convex demixing, with applications. *Foundations of Computational Mathematics*, 14(3):503–567, 2014.
- [166] B. McWilliams and G. Montana. Subspace clustering of high-dimensional data: a predictive approach. *Arxiv preprint arXiv:1203.1065*, 2012.
- [167] N. Meinshausen and P. Bühlmann. High-dimensional graphs and variable selection with the lasso. *The Annals of Statistics*, 34(3):1436–1462, 2006.
- [168] J. Miao, J. E. Amonette, Y. Nishino, T. Ishikawa, and K. O. Hodgson. Direct determination of the absolute electron density of nanostructured and disordered materials at sub-10-nm resolution. *Physical Review B*, 68(1):012201, 2003.
- [169] J. Miao, P. Charalambous, J. Kirz, and D. Sayre. Extending the methodology of x-ray crystallography to allow imaging of micrometre-sized non-crystalline specimens. *Nature*, 400(6742):342–344, 1999.
- [170] J. Miao, C. Chen, C. Song, Y. Nishino, Y. Kohmura, T. Ishikawa, D. Ramunno-Johnson, T. Lee, and S. H. Risbud. Three-dimensional gan-ga 2 o 3 core shell structure revealed by x-ray diffraction microscopy. *Physical review letters*, 97(21):215503, 2006.
- [171] J. Miao, K. O. Hodgson, T. Ishikawa, C. A. Larabell, M. A. LeGros, and Y. Nishino. Imaging whole escherichia coli bacteria by using single-particle x-ray diffraction. *Proceedings of the National Academy of Sciences*, 100(1):110–112, 2003.
- [172] J. Miao, K. O. Hodgson, and D. Sayre. An approach to three-dimensional structures of biomolecules by using single-molecule diffraction images. *Proceedings of the National Academy of Sciences*, 98(12):6641–6645, 2001.
- [173] J. Miao, T. Ishikawa, B. Johnson, E. H. Anderson, B. Lai, and K. O. Hodgson. High resolution 3D x-ray diffraction microscopy. *Physical review letters*, 89(8):088303, 2002.

- [174] J. Miao, T. Ishikawa, Q. Shen, and T. Earnest. Extending x-ray crystallography to allow the imaging of noncrystalline materials, cells, and single protein complexes. *Annu. Rev. Phys. Chem.*, 59:387–410, 2008.
- [175] J. Miao, Y. Nishino, Y. Kohmura, B. Johnson, C. Song, S. H. Risbud, and T. Ishikawa. Quantitative image reconstruction of gan quantum dots from oversampled diffraction intensities alone. *Physical review letters*, 95(8):085503, 2005.
- [176] J. Miao, T. Ohsuna, O. Terasaki, K. O. Hodgson, and M. A. OKeefe. Atomic resolution three-dimensional electron diffraction microscopy. *Physical review letters*, 89(15):155502, 2002.
- [177] R. P. Millane. Phase retrieval in crystallography and optics. *JOSA A*, 7(3):394–411, 1990.
- [178] V. D. Milman and G. Schechtman. *Asymptotic Theory of Finite Dimensional Normed Spaces: Isoperimetric Inequalities in Riemannian Manifolds*, volume 1200. Springer, 1986.
- [179] D. Mondragon and V. Voroninski. Determination of all pure quantum states from a minimal number of observables. *arXiv preprint arXiv:1306.1214*, 2013.
- [180] Y. Mroueh. Robust phase retrieval and super-resolution from one bit coded diffraction patterns. *arXiv preprint arXiv:1402.2255*, 2014.
- [181] Y. Mroueh and L. Rosasco. Quantization and greed are good: One bit phase retrieval, robustness and greedy refinements. *arXiv preprint arXiv:1312.1830*, 2013.
- [182] K. G. Murty and S. N. Kabadi. Some NP-complete problems in quadratic and nonlinear programming. *Mathematical programming*, 39(2):117–129, 1987.
- [183] S. Nam, M. E. Davies, M. Elad, and R. Gribonval. Cosparse analysis modeling-uniqueness and algorithms. In *Acoustics, Speech and Signal Processing*

- (ICASSP), 2011 IEEE International Conference on, pages 5804–5807. IEEE, 2011.
- [184] S. Nam, M. E. Davies, M. Elad, and R. Gribonval. The cosparse analysis model and algorithms. *Applied and Computational Harmonic Analysis*, 34(1):30–56, 2013.
 - [185] A. Nemirovski. Lectures on modern convex optimization. In *Society for Industrial and Applied Mathematics (SIAM)*. Citeseer, 2001.
 - [186] Y. Nesterov. *Introductory lectures on convex optimization*, volume 87 of *Applied Optimization*. Kluwer Academic Publishers, Boston, MA, 2004. A basic course.
 - [187] P. Netrapalli, P. Jain, and S. Sanghavi. Phase retrieval using alternating minimization. *arXiv preprint arXiv:1306.0160*, 2013.
 - [188] R. Neutze, R. Wouts, D. van der Spoel, E. Weckert, and J. Hajdu. Potential for biomolecular imaging with femtosecond x-ray pulses. *Nature*, 406(6797):752–757, 2000.
 - [189] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 2:849–856, 2002.
 - [190] Y. Nishino, J. Miao, and T. Ishikawa. Image reconstruction of nanostructured nonperiodic objects only from oversampled hard x-ray diffraction intensities. *Physical Review B*, 68(22):220101, 2003.
 - [191] K. A. Nugent, A. G. Peele, H. N. Chapman, and A. P. Mancuso. Unique phase recovery for nonperiodic objects. *Physical review letters*, 91(20):203902, 2003.
 - [192] H. Ohlsson, A. Y. Yang, R. Dong, and S. S. Sastry. Compressive phase retrieval from squared output measurements via semidefinite programming. *arXiv preprint arXiv:1111.6323*, 2011.

- [193] P. Orlik. *Introduction to arrangements*, volume 72. American Mathematical Soc., 1989.
- [194] S. Oymak and B. Hassibi. Sharp mse bounds for proximal denoising. *arXiv preprint arXiv:1305.2714*, 2013.
- [195] S. Oymak, A. Jalali, M. Fazel, Y. C. Eldar, and B. Hassibi. Simultaneously structured models with application to sparse and low-rank matrices. *arXiv preprint arXiv:1212.3753*, 2012.
- [196] N. Ozay, M. Sznaier, and C. Lagoa. Model (in) validation of switched arx systems with unknown switches and its application to activity monitoring. In *IEEE Conference on Decision and Control, CDC 2010.*, pages 7624–7630.
- [197] N. Ozay, M. Sznaier, C. Lagoa, and O. Camps. GPCA with denoising: A moments-based convex approach. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2010.*, pages 3209–3216. IEEE.
- [198] L. Parsons, E. Haque, and H. Liu. Subspace clustering for high dimensional data: a review. *ACM SIGKDD Explorations Newsletter*, 6(1):90–105, 2004.
- [199] M. A. Pfeifer, G. J. Williams, I. A. Vartanyants, R. Harder, and I. K. Robinson. Three-dimensional mapping of a deformation field inside a nanocrystal. *Nature*, 442(7098):63–66, 2006.
- [200] H. M. Quiney, A. G. Peele, Z. Cai, D. Paterson, and K. A. Nugent. Diffractive imaging of highly focused x-ray fields. *Nature Physics*, 2(2):101–104, 2006.
- [201] P. A. Randall. *Sparse recovery via convex optimization*. PhD thesis, California Institute of Technology, 2009.
- [202] S. Rangan. Generalized approximate message passing for estimation with random linear mixing. In *Information Theory Proceedings (ISIT), 2011 IEEE International Symposium on*, pages 2168–2172. IEEE, 2011.

- [203] J. Ranieri, A. Chebira, Y. M. Lu, and M. Vetterli. Phase retrieval for sparse signals: Uniqueness conditions. *arXiv preprint arXiv:1308.3058*, 2013.
- [204] O. Raz, N. Dudovich, and B. Nadler. Vectorial phase retrieval of 1-D signals. 2013.
- [205] H. Reichenbach. *Philosophic foundations of quantum mechanics*. University of California Pr, 1965.
- [206] I. K. Robinson, I. A. Vartanyants, G. J. Williams, M. A. Pfeifer, and J. A. Pitney. Reconstruction of the shapes of gold nanocrystals using coherent x-ray diffraction. *Physical review letters*, 87(19):195505, 2001.
- [207] J. M. Rodenburg. Ptychography and related diffractive imaging methods. *Advances in Imaging and Electron Physics*, 150:87–184, 2008.
- [208] M. Rosenbaum and A. B. Tsybakov. Sparse recovery under matrix uncertainty. *The Annals of Statistics*, 38(5):2620–2651, 2010.
- [209] M. Rosenbaum and A. B. Tsybakov. Improved matrix uncertainty selector. In *From Probability to Statistics and Back: High-Dimensional Models and Processes—A Festschrift in Honor of Jon A. Wellner*, pages 276–290. Institute of Mathematical Statistics, 2013.
- [210] J. L. C. Sanz, T. S. Huang, and T. Wu. A note on iterative Fourier transform phase reconstruction from magnitude. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 32(6):1251–1254, 1984.
- [211] D. Sayre. Some implications of a theorem due to shannon. *Acta Crystallographica*, 5(6):843–843, 1952.
- [212] P. Schniter and S. Rangan. Compressive phase retrieval via generalized approximate message passing. In *Communication, Control, and Computing (Allerton), 2012 50th Annual Allerton Conference on*, pages 815–822. IEEE, 2012.

- [213] P. Shah, B. N. Bhaskar, G. Tang, and B. Recht. Linear system identification via atomic norm regularization. *arXiv preprint arXiv:1204.0590*, 2012.
- [214] D. Shapiro, P. Thibault, T. Beetz, V. Elser, M. Howells, C. Jacobsen, J. Kirz, E. Lima, H. Miao, A. M. Neiman, et al. Biological imaging by soft x-ray diffraction microscopy. *Proceedings of the National Academy of Sciences of the United States of America*, 102(43):15343–15346, 2005.
- [215] Y. Shechtman, Y. C. Eldar, O. Cohen, H. N. Chapman, J. Miao, and M. Segev. Phase retrieval with application to optical imaging. *arXiv preprint arXiv:1402.7350*, 2014.
- [216] Y. Shechtman, Y. C. Eldar, A. Szameit, and M. Segev. Sparsity based subwavelength imaging with partially incoherent light via quadratic compressed sensing. *arXiv preprint arXiv:1104.4406*, 2011.
- [217] M. Soltanolkotabi and E. J. Candès. A geometric analysis of subspace clustering with outliers. *The Annals of Statistics*, 40(4):2195–2238, 2012.
- [218] M. Soltanolkotabi, E. Elhamifar, and E. J. Candès. Robust subspace clustering. *Annals of Statistics*, 42(2):669–699, 2014.
- [219] G. Tang, B. N. Bhaskar, and B. Recht. Near minimax line spectral estimation. In *Information Sciences and Systems (CISS), 2013 47th Annual Conference on*, pages 1–6. IEEE, 2013.
- [220] G. Tang, B. N. Bhaskar, P. Shah, and B. Recht. Compressed sensing off the grid. *arXiv preprint arXiv:1207.6053*, 2012.
- [221] P. Thibault, M. Dierolf, O. Bunk, A. Menzel, and F. Pfeiffer. Probe retrieval in ptychographic coherent diffractive imaging. *Ultramicroscopy*, 109(4):338–343, 2009.
- [222] P. Thibault, V. Elser, C. Jacobsen, D. Shapiro, and D. Sayre. Reconstruction of a yeast cell from x-ray diffraction data. *Acta Crystallographica Section A: Foundations of Crystallography*, 62(4):248–261, 2006.

- [223] M.E. Tipping and C.M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622, 1999.
- [224] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision*, 9(2):137–154, 1992.
- [225] R. Tron and R. Vidal. A benchmark for the comparison of 3-d motion segmentation algorithms. In *Computer Vision and Pattern Recognition, 2007. CVPR’07. IEEE Conference on*, pages 1–8. IEEE, 2007.
- [226] J. A. Tropp. User-friendly tail bounds for sums of random matrices. *Foundations of Computational Mathematics*, 12(4):389–434, 2012.
- [227] P. Tseng. Nearest q-flat to m points. *Journal of Optimization Theory and Applications*, 105(1):249–252, 2000.
- [228] Z. Uo, W. Ma, A. C. So, Y. Ye, and S. Zhang. Semidefinite relaxation of quadratic optimization problems. *IEEE Signal Processing Magazine*, 27(3):20–34, 2010.
- [229] users.ece.gatech.edu/~sasif/homotopy.
- [230] S. Vaiter, G. Peyre, C. Dossal, and J. Fadili. Robust sparse analysis regularization. *Information Theory, IEEE Transactions on*, 59(4):2001–2016, 2013.
- [231] R. Vershynin. Lectures in geometric functional analysis. *Unpublished manuscript. Available at <http://www-personal.umich.edu/~romanv/papers/GFA-book/GFA-book.pdf>*, 2011.
- [232] R. Vershynin. Introduction to the non-asymptotic analysis of random matrices. Chapter 5 of the book Compressed Sensing, Theory and Applications, ed. Y. Eldar and G. K. 2012.

- [233] R. Vidal. Subspace clustering. *IEEE Signal Processing Magazine.*, 28(2):52–68, 2011.
- [234] R. Vidal, Y. Ma, and S. Sastry. Generalized Principal Component Analysis (GPCA). *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(12):1945–1959, 2005.
- [235] A. Vogt. Position and momentum distributions do not determine the quantum mechanical state. 1978.
- [236] U. Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.
- [237] I. Waldspurger, A. d’Aspremont, and S. Mallat. Phase recovery, maxcut and complex semidefinite programming. *arXiv preprint arXiv:1206.0102*, 2012.
- [238] A. Walther. The question of phase retrieval in optics. *Journal of Modern Optics*, 10(1):41–49, 1963.
- [239] F. H. C. Watson, J. D. and Crick. A structure for deoxyribose nucleic acid. *Nature*, 171, 1953.
- [240] S. Weigert. Simple minimal informationally complete measurements for qudits. *International Journal of Modern Physics B*, 20(11n13):1942–1955, 2006.
- [241] Z. Wen, C. Yang, X. Liu, and S. Marchesini. Alternating direction methods for classical and ptychographic phase retrieval. *Inverse Problems*, 28(11):115010, 2012.
- [242] G. J. Williams, M. A. Pfeifer, I. A. Vartanyants, and I. K. Robinson. Three-dimensional imaging of microstructure in au nanocrystals. *Physical review letters*, 90(17):175501, 2003.
- [243] G. J. Williams, H. M. Quiney, B. B. Dhal, C. Q. Tran, K. A. Nugent, A. G. Peele, D. Paterson, and M. D. De Jonge. Fresnel coherent diffractive imaging. *Physical review letters*, 97(2):025506, 2006.

- [244] P. Wojtaszczyk. Stability and instance optimality for Gaussian measurements in compressed sensing. *Foundations of Computational Mathematics*, 10(1):1–13, 2010.
- [245] Y. Wu, Z. Zhang, T. S. Huang, and J. Y. Lin. Multibody grouping via orthogonal subspace decomposition. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 2, pages II–252. IEEE, 2001.
- [246] X. Xiao and Q. Shen. Wave propagation and phase retrieval in fresnel diffraction by a distorted-object approach. *Physical Review B*, 72(3):033103, 2005.
- [247] J. Yan and M. Pollefeys. A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate. In *Computer Vision–ECCV 2006*, pages 94–106. Springer, 2006.
- [248] J. Yan and M. Pollefeys. A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate. *ECCV 2006*, pages 94–106, 2006.
- [249] G. Yang, B. Dong, B. Gu, J. Zhuang, and O. K. Ersoy. Gerchberg-Saxton and Yang-Gu algorithms for phase retrieval in a nonunitary transform system: a comparison. *Applied optics*, 33(2):209–218, 1994.
- [250] A. Zhang, N. Fawaz, S. Ioannidis, and A. Montanari. Guess who rated this movie: Identifying users through subspace clustering. *Arxiv preprint arXiv:1208.1544*, 2012.
- [251] T. Zhang, A. Szlam, and G. Lerman. Median k -flats for hybrid linear modeling with many outliers. In *IEEE International Conference on Computer Vision Workshops, ICCV 2009.*, pages 234–241.
- [252] T. Zhang, A. Szlam, Y. Wang, and G. Lerman. Hybrid linear modeling via local best-fit flats. *International Journal of Computer Vision*, pages 1–24, 2012.

- [253] F. Zhou, F. Torre, and J. K. Hodgins. Aligned cluster analysis for temporal segmentation of human motion. In *IEEE International Conference on Automatic Face and Gesture Recognition, FG 2008.*, pages 1–7.
- [254] J. M. Zuo, I. Vartanyants, M. Gao, R. Zhang, and L. A. Nagahara. Atomic resolution imaging of a carbon nanotube from diffraction intensities. *Science*, 300(5624):1419–1421, 2003.