HOMEWORK5

March 13, 2021

ZHA Mengyue
20668901
HKUST
MATH Department

Contents

0.1 Brief Introduction

The blue hyperlinks are available!!! Since there are a lot of scripts for this homework, I upload this homework toMy Solutions to HW5 Github. It's convenient to download or check them in Github.
Here is the structure of solutions

1. The solution for Problem1
   (a) RPCA.m is a very basic realization of Robust PCA. It answers the question a of Problem1
   (b) RecoveryWithP.m is a script explore the probability of successful recover when given different parameters $p \in [0,1]$.
   (c) RecoveryWithR.m is a script explore the probability of successful recover given different parameter r.
   (d) RecoveryWithPandR.m is a script explore the phase transition wrt p and r. I also plot the graph of phase transition in this scrip. The outcome is saved as PhaseTransition.fig
2. All solutions to the second problem are saved in the scrip SparsePCA.m in the subfolder SPCA.
3. All solutions to the third problem are saved in the folder SNLSDP. Since this problem is complicated we will deter the introduction to the section Problem3.

0.2 Problem1

1. for simplicity we do 20 trials for each setting of hyper-parameter to calculate the recovry probability
2. the probability for $r = 1:20$ is $prob = [0.4000, 0.1400, 0.0800, 0, \cdots, 0]$ where is is set to be 0.9 by default
3. the probability for $p = 0.1:0.1:1$ is $prob = [0, \cdots, 0, 0.4]$

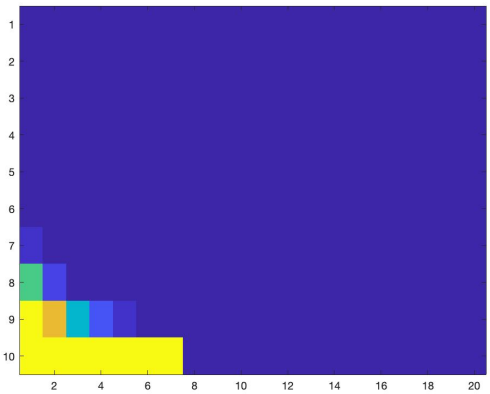

Figure 1: Phase Transition

Conclusions

1. There is a clear phase transition in the Robust PCA Algorithm.
2. We have higher probability to successfully recover the data when r is small while p is large

0.3 Problem2

The basic outcomes of a,b,c can be easily find once run this fast script SparsePCA.m. Thus, we only post part of the graph outcomes of d. The full outcome can be find in my Github.

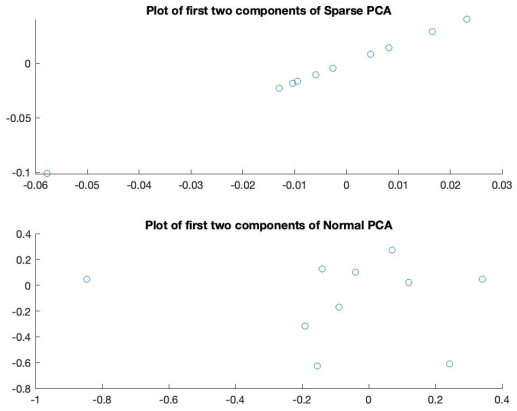

Figure 2: Comparison of Sparse PCA and Classical PCA with no extraction

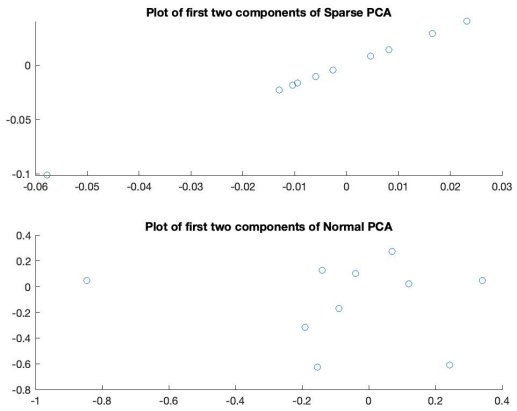

Figure 3: Comparison of Sparse PCA and Classical PCA with once extraction

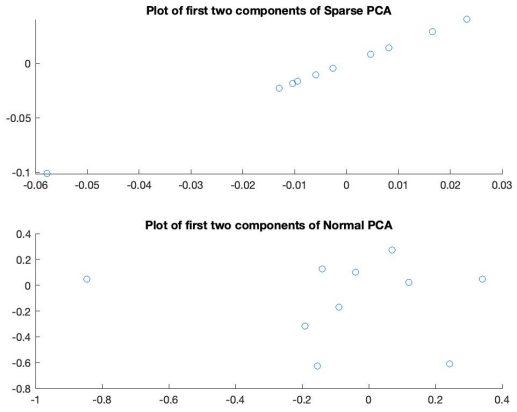

Figure 4: Comparison of Sparse PCA and Classical PCA with twice extractions

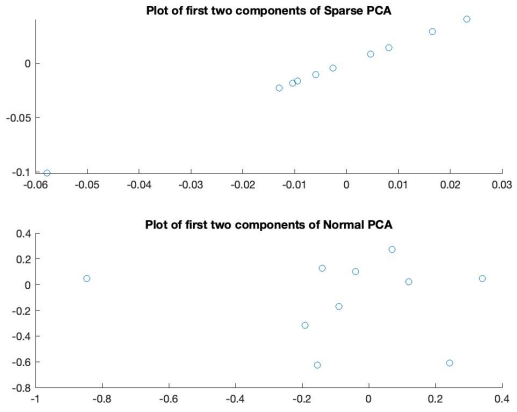

Figure 5: Comparison of Sparse PCA and Classical PCA with three extractions

Conclusions Sparse PCA is consistently better than Classical PCA in both the experiments of changing λ and doing extractions respectively.

0.4 Problem3

SNLSDP Structure

1. ProteinRecovry.m

   ProteinRecovry.m is the main function for this problem.

   Once you want the change the dataset we are experimenting on, just change the file read consoles between line 56 and 65. All the data sets mentioned in the notation can be find under the SNLSDP.

2. The folder figs

   The folder figs contains outcomes and related explanatory pictures from the internet.

3. starup.m

   starup.m is used to start up

4. A SDPT3-4.0 package

   SDPT3-4.0 package for solving convex problems which need to be installed. The detail and other things about this folder is listed below.

Remark on SDPT3-4.0

1. All contents under the original SDPT folder under SNLSDP provided by Kim-Chuan Toh, Pratik Biswas, and Yinyu Ye, are replaced by the latest version of SDPT3. Otherwise, enumerate errors will be raised.

2. you need to enter the folder SDPT3-4.0 and run install_sdpt3.m to install this package

Figure Outcomes for Protein 1WVN

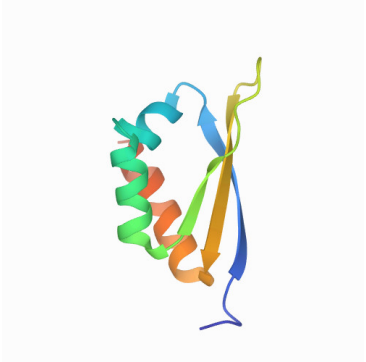

Figure 6: The real image of Protein 1WVN

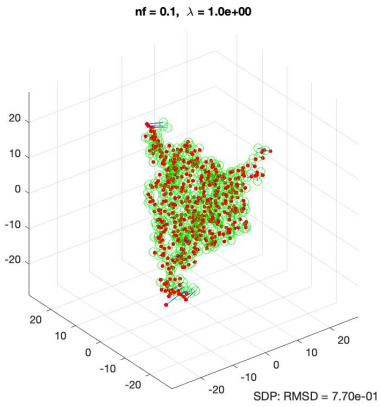

Figure 7: The first recovered image of Protein 1WVN

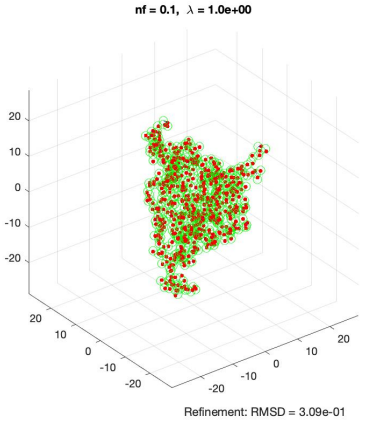

Figure 8: The second recovered image of Protein 1WVN

Figure Outcomes for Protein 2HDA

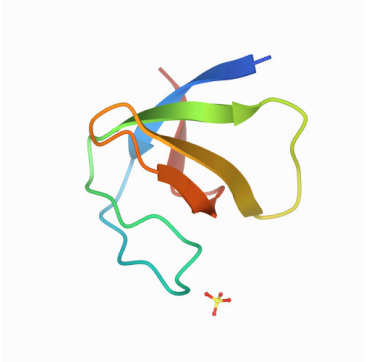

Figure 9: The real image of Protein 2HDA

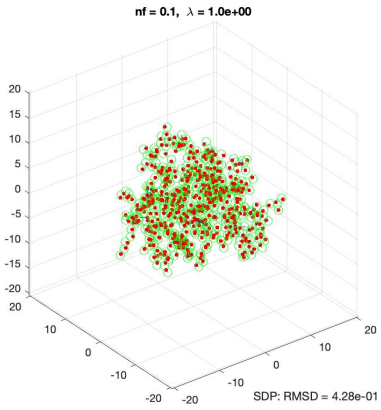

Figure 10: The first recovered image of Protein 2HDA

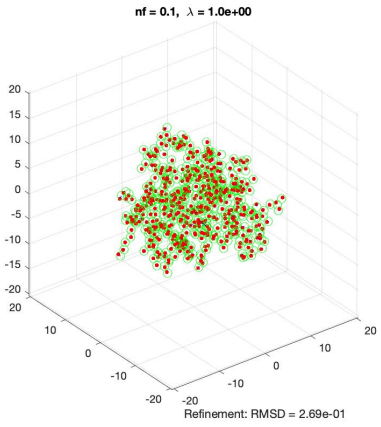

Figure 11: The second recovered image of Protein 2HDA

Figure Outcomes for Protein 1GM2

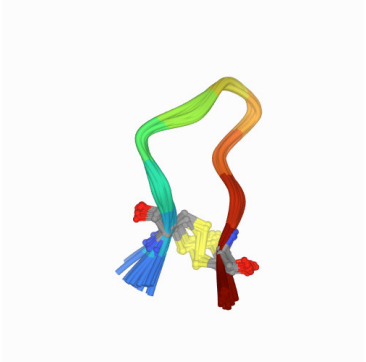

Figure 12: The real image of Protein 1GM2

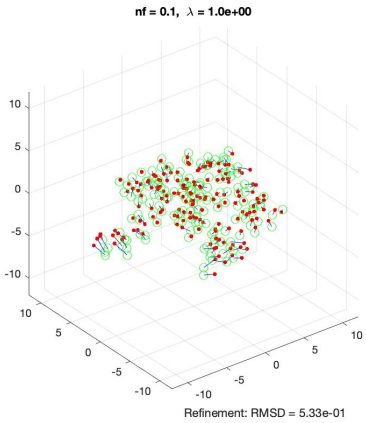

Figure 13: The first recovered image of Protein 1GM2

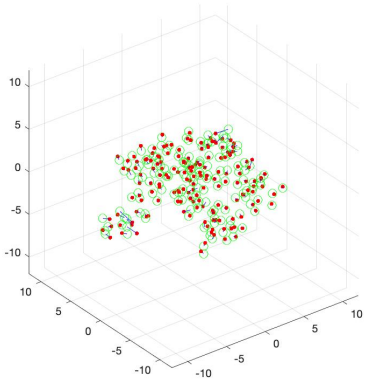

Figure 14: The second recovered image of Protein 1GM2

Conclusions

1. All the answer were set under radius equals 8. For complex protein small radius let sdp unsolvable.

2. SNLSDP suit for simple proteins like 1gm2. I tried to apply the SNLSDP on covid-19's structure but failed since it's structure is too complicated listed below.

3. Also the input sequence shuold not be too short. For example, the data 1R9H.pdb in Kirill Konovalov's package is not runnable actually. The Xopt variable in the main function will eventually degenerate to 0.

4. According to my shallow experience. A typical protein that is solvable under SNLSDP should better have number of sensor between 100 to 500. number of constraints from 5000-18000