ECE 269: Linear Algebra and Applications Winter 2023

Mini Project # 2 Report

By Abhijith Srinivasachar Dasharathi (A59020719)

Finding Sparse Solutions via Orthogonal Matching Pursuit (OMP)

In this mini project, we will implement and study the performance of the Orthogonal Matching Pursuit (OMP) algorithm for recovering sparse signals and images.

Consider the measurement model.

$$y = Ax + n$$

where $y \in R^M$ is the (compressed, $M < N$) measurement, $A \in R^{M\times N}$ is the measurement matrix, and $n \in R^M$ is the additive noise. Here, $x \in R^N$ is the unknown signal (to be estimated) with $s \ll N$ non-zero elements. The indices of the non-zero entries of x (also known as the support of x) is denoted by $S = \{i|x_i \neq 0\}$, with $|S|=s$.

1. Performance Metrics: Let x^ be the estimate of x obtained from OMP. To measure the performance of OMP, we consider the Normalized Error defined as

$$\frac{\| x - \hat{x} \|_2}{\| x \|_2}$$

The average Normalized Error is obtained by averaging the Normalized Error over 2000 Monte Carlo runs.

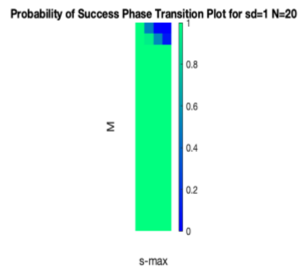
2. Experimental setup:

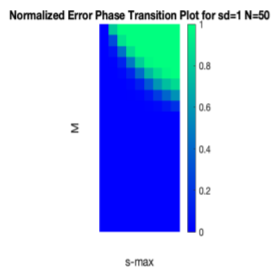
a) Generate A as a random matrix with independent and identically distributed entries drawn from the standard normal distribution. Normalize the columns of A.
b) Generate the sparse vector x with random support of cardinality s (i.e. s indices are generated uniformly at random from integers 1 to N), and non-zero entries drawn as uniform random variables in the range $[-10, -1] \cup [1, 10]$.
c) The entries of noise n are drawn independently from the normal distribution with standard deviation σ and zero mean.
d) For each cardinality $s \in [1, s_{max}]$, the average Normalized Error should be computed by repeating step (a) to step (c) 2000 times and averaging the results over these 2000 Monte Carlo runs.

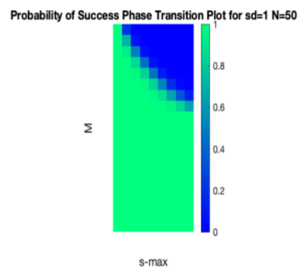
3. Noiseless case: ($n = 0$)

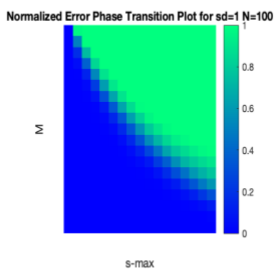
Implement OMP (you may stop the OMP iterations once $\|y - Ax^{(k)}\|_2$ is close to 0 and evaluate its performance. Calculate the probability of Exact Support Recovery (i.e., the fraction of runs when $\hat{S} = S$) by averaging over 2000 random realizations of A, as a function of M and s_{max} (for different fixed values of N). For each N, the probability of exact support recovery is a two-dimensional plot (function of M and s_{max}) and you can display it as an image. The resulting plot is called the “noiseless phase transition" plot, and it shows how many measurements (M) are needed for OMP to successfully recover the sparse signal, as a function of s_{max}. Do you observe a sharp transition region where the probability quickly transitions from a large value (close to 1) to a very small value (close to 0)? Generate different phase transition plots for the following values of N: 20, 50 and 100. Regenerate phase transition plots for average Normalized Error (instead of probability of successful recovery). Comment on both kinds of plots.

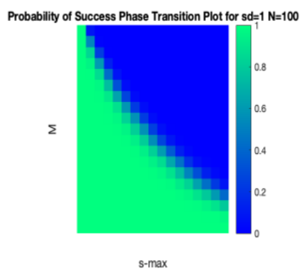
Solution: Yes, I do observe a sharp transition region where the probability quickly transitions from a large value (close to 1) to a very small value (close to 0). This can be observed in the attached plots in the following page. When I regenerate the transition plot for Normalized error, the similar transition is observed but its reverse in values as probability of success correspond to zero error. The probability plot is sharper compared to normalized error plot and can be clearly observed for N=100.





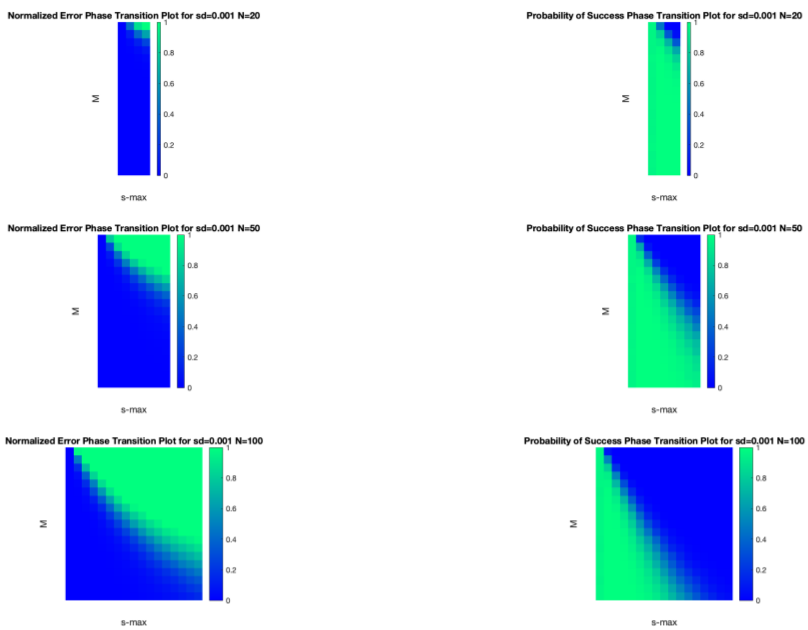


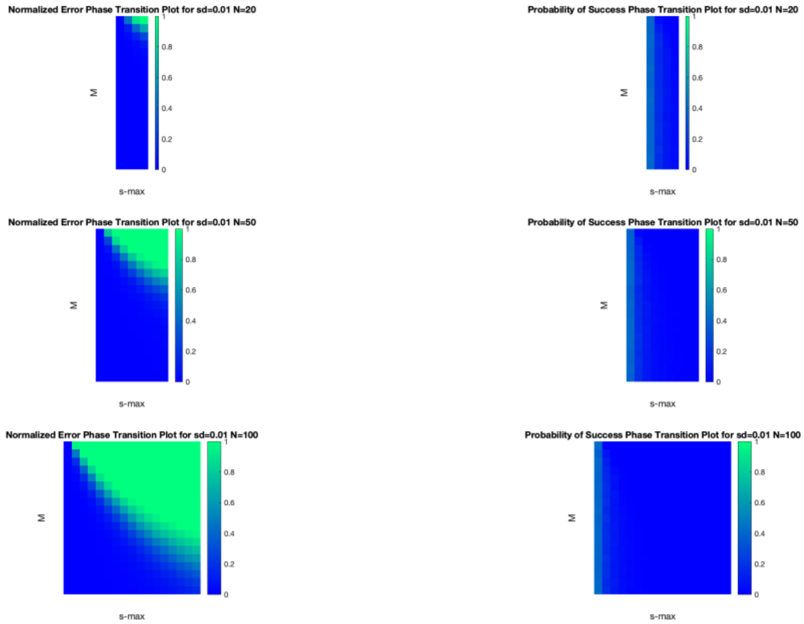




4. Noisy case: ($n \neq 0$)
   a) Assume that sparsity s is known. Implement OMP (terminate the algorithm after first s columns of A are selected). Generate "noisy phase transition" plots (for fixed N and σ) where success is defined as the event that the Normalized Error is less than 10^{-3}. Repeat the experiment for two values of σ (one small and one large) and choose N as 20, 50, and 100. Comment on the results.

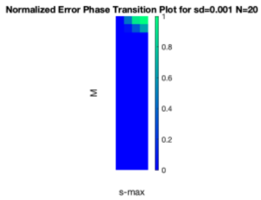
   Solution: I observe that when σ is small, there is a sharp transition in probability plot while for bigger σ, the probability plot is always 0 and doesn't change. But normalized error plot is always showing transition with varying σ. For higher N, the sharpness of transition increases compared to lower N. When σ is closer to the normalized error threshold, the probability plot shows sharp transitions and when it is bigger in the order of magnitude, then the transition is lost. Below the first plot is for smaller σ and the next is for bigger σ.

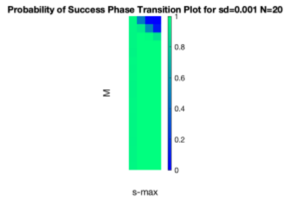


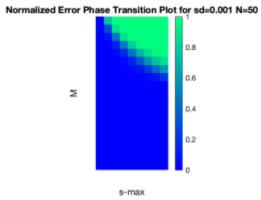


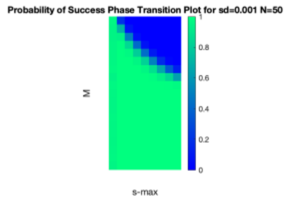
b) Assume the sparsity s is NOT known, but $\|n\|_2$ is known. Implement OMP where you may stop the OMP iterations once $\|y - Ax^{(k)}\| \leq \|n\|_2$). Generate phase transition plots using the same criterion for success as the previous part. Comment on the results.

Solution: I observe the similar behavior as part a here. When σ is small, there is a sharp transition in probability plot while for bigger σ, the probability plot is always 0 and doesn't change. But normalized error plot is always showing transition with varying σ. For higher N, the sharpness of transition increases compared to lower N. When σ is closer to the normalized error threshold, the probability plot shows sharp transitions and when it is bigger in the order of magnitude, then the transition is lost. This shows us that just knowing the l2 norm of noise is enough for OMP to work well. Below the first plot is for smaller σ and the next is for bigger σ.

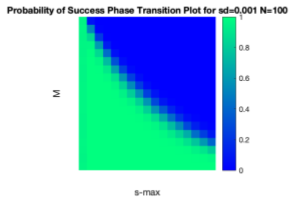

Normalized Error Phase Transition Plot for sd=0.001 N=20
M
s-max
1
0.8
0.6
0.4
0.2
0

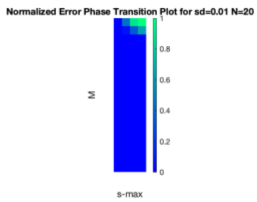

Probability of Success Phase Transition Plot for sd=0.001 N=20
M
s-max
1
0.8
0.6
0.4
0.2
0

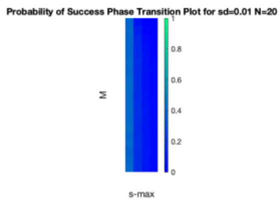

Normalized Error Phase Transition Plot for sd=0.001 N=50
M
s-max
1
0.8
0.6
0.4
0.2
0

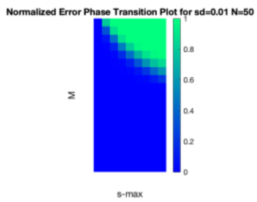

Probability of Success Phase Transition Plot for sd=0.001 N=50
M
s-max
1
0.8
0.6
0.4
0.2
0

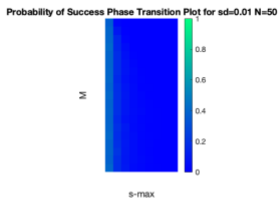

Normalized Error Phase Transition Plot for sd=0.001 N=100
M
s-max
1
0.8
0.6
0.4
0.2
0

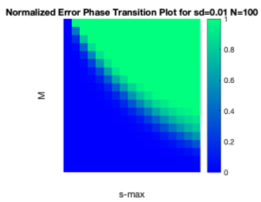

Probability of Success Phase Transition Plot for sd=0.001 N=100
M
s-max
1
0.8
0.6
0.4
0.2
0


Normalized Error Phase Transition Plot for sd=0.01 N=20
M
s-max
1
0.8
0.6
0.4
0.2
0


Probability of Success Phase Transition Plot for sd=0.01 N=20
M
s-max
1
0.8
0.6
0.4
0.2
0


Normalized Error Phase Transition Plot for sd=0.01 N=50
M
s-max
1
0.8
0.6
0.4
0.2
0


Probability of Success Phase Transition Plot for sd=0.01 N=50
M
s-max
1
0.8
0.6
0.4
0.2
0


Normalized Error Phase Transition Plot for sd=0.01 N=100
M
s-max
1
0.8
0.6
0.4
0.2
0


Probability of Success Phase Transition Plot for sd=0.01 N=100
M
s-max
1
0.8
0.6
0.4
0.2
0

5. Decode a Compressed Message: In this part of the assignment, you will uncover a hidden message from their compressed sketches (generated using random measurement matrices) using the OMP algorithm that seeks the sparsest solution. An unknown and sparse image X, containing a message, has been compressed using three different random matrices (of different sizes) A_1, A_2, A_3 to produce three corrupted images as follows.

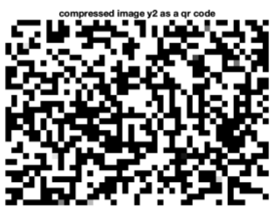
$$Y_i = A_i X$$

The corrupted images and the measurement matrices can be accessed from

https://drive.google.com/drive/folders/1DqRlGtFo7Ytw_A2TS5vzS5twJJOvcXan?usp=sharing

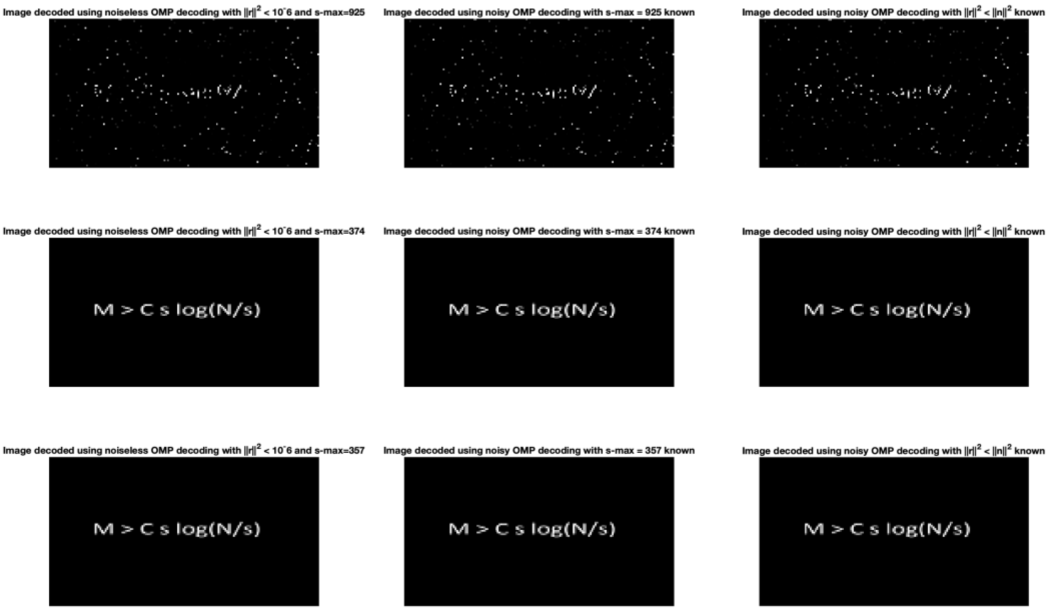
a) Can you guess the message by simply displaying the compressed images?

   Solution: No, in fact the image looks like a QR code, and we cannot make any message out of it. Please find the attached image of compressed image.



b) Using OMP, recover X from Y_1, Y_2, Y_3. Figure out the appropriate stopping criteria. Reshape the recovered X into a 2D image of size 90×160 and decode the message. Show your results for each case. Compare these results with the Least Squares Solution.

Solution: Please find the decoded images for the message. In the below figure, 1st row corresponds to least square solutions of A1, y1. 2nd row corresponds to least squares solutions of A2, y2. 3rd row corresponds to least squares solutions of A3, y3. To identify s-max and n values the appropriate stopping criteria is $\|y - Ax^{(k)}\|_2 < 10^{-6}$. From this we get recovered images. The least squares solution for A1, y1 is very bad and is very far from the actual message while the least squares solution for A2, y2 and A3, y3 are very close to original message.



c) Which corrupted image gave you the best result for OMP? Can you explain why?

Solution: The corrupted image y2 and y3 both gave best result. This is because the M value for both these images are greater than the minimum required size of M as indicated by the decoded message.

d) (For Fun) Can you make an (educated) guess about the meaning of this message?

Solution: The message says "M > C s log(N/s)". This means that for the OMP to recover any compressed message, the M should be at least some constant "C" times the expression "s log(N/s)".

We observe N=14400 and the corresponding M and s for all 3 cases are shown below.

	A1, y1	A2, y2	A3, y3
N	14400	14400	14400
M	960	1440	2880
S_max	925	374	357
s log(N/s)	2539.3	1365.4	1319.9
C (M/ s log(N/s))	0.3781	1.0546	2.1820

From above table we observe that the value of C lies within $0.3781 < C <= 1.0546$.

Just a smart guess for C would be $C \cong 1$. Then for N=14400 and s = 350 (assumption) those y_i compressed with M > s log(N/s) (M > 1301) can recover x_i using OMP.