ECE 269 - Final Project Report
Orthogonal Matching Pursuit

Eric Weise

December 17, 2020

My project is hosted on GitHub:
`https://github.com/ericdweise/ece269/blob/master/report.pdf`

Background

Noiseless Signal Recovery (*part c*)

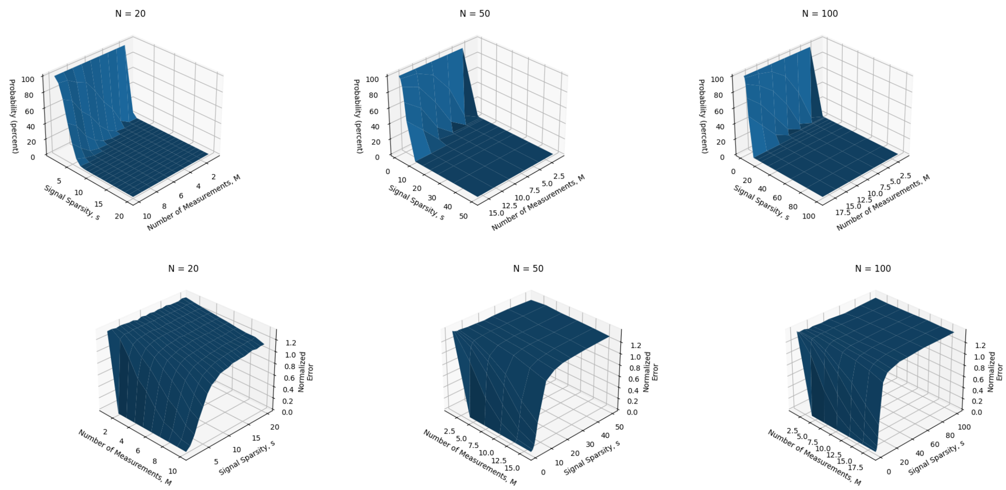

Figure 1: Noiseless support set recovery. Top row: The probability of Exact Support Recovery. Bottom row: Normalized error of recovered signal. NOTE: The ESR and the error plots are viewed from different angles.

The Exact Signal Recovery (ESR) rate is very highly correlated to s, and somewhat correlated to M. This is evident in the sharp

The Normalized Error plots show roughtly the inverse trend as that in the ESR plots: The Error rate is low for small sparsity, and reach a plateau close to 1.2.

Noisy Signal Recovery With Known Sparsity (*Part d-i*)

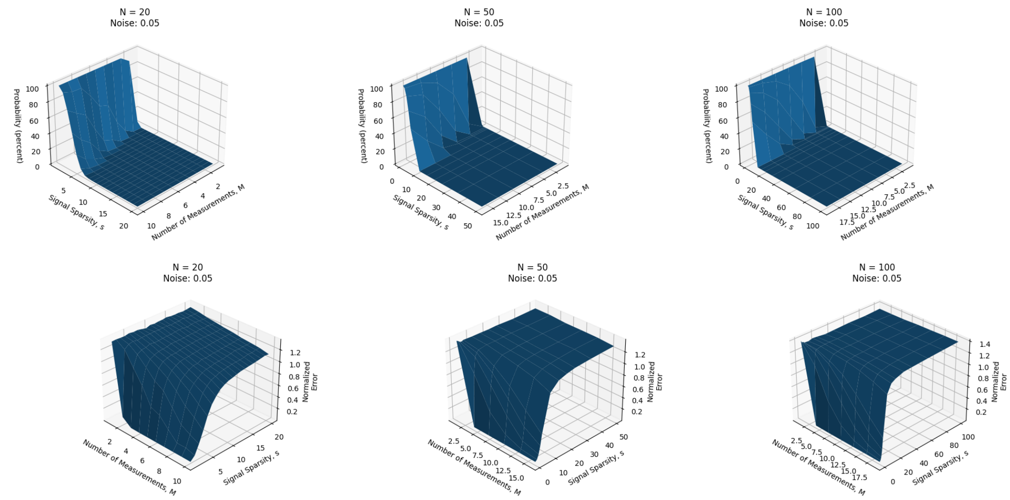

Figure 2: Noisy support set recovery **variance=0.05**. Stop condition achieved when loop count is the sparsity of the input signal. Top row: The probability of Exact Support Recovery. Bottom row: Normalized error of recovered signal.

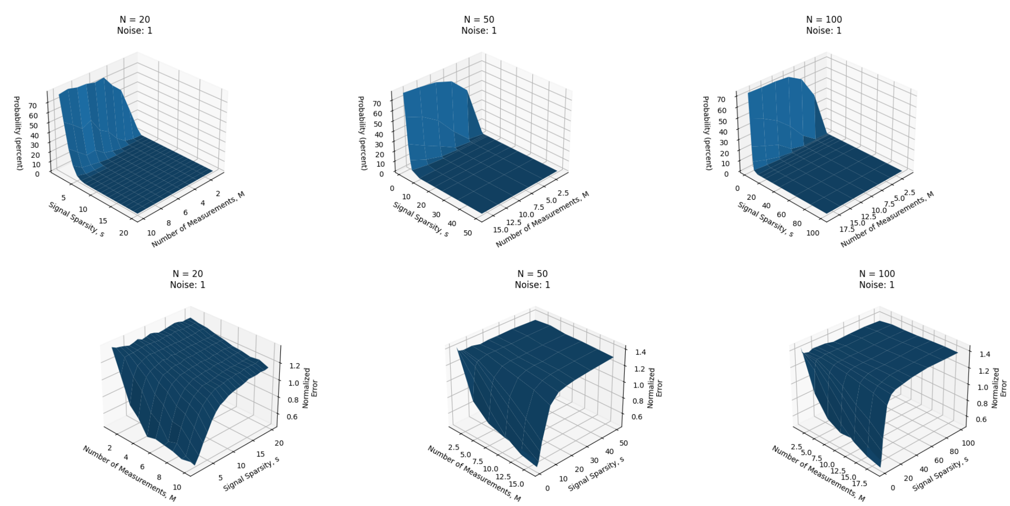

Figure 3: Noisy support set recovery **variance=1**. Stop condition achieved when loop count is the sparsity of the input signal. Top row: The probability of Exact Support Recovery. Bottom row: Normalized error of recovered signal.

As we might predict, adding small amounts of noise changes the ESR and Normalized Error plots slightly, while adding more noise washes out the lots more significantly.

Noisy Signal Recovery (*part d-ii*)

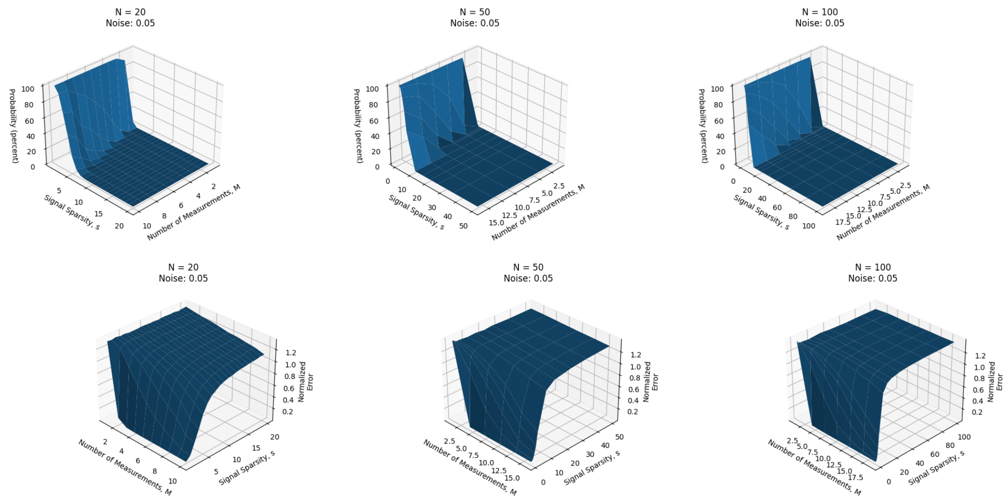

Figure 4: Noisy support set recovery **variance=0.05**. Stop condition achieved when the 2-norm of residual is less than 0.001. Top row: The probability of Exact Support Recovery. Bottom row: Normalized error of recovered signal.

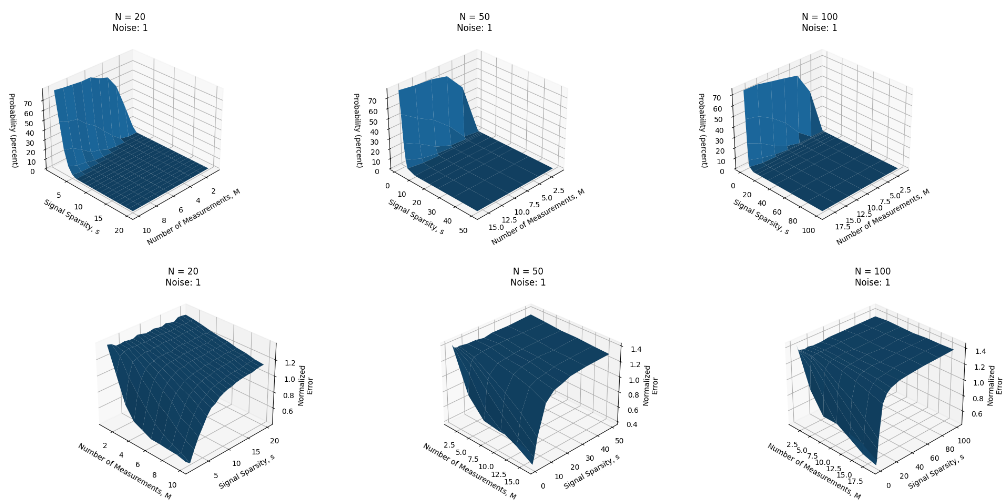

Figure 5: Noisy support set recovery **variance=1**. Stop condition achieved when the 2-norm of residual is less than 0.001. Top row: The probability of Exact Support Recovery. Bottom row: Normalized error of recovered signal.

Image Recovery (*Part d-iii*)

Compressive Sampling and Signal Recovery Methods

Compressive sensing is a rich field. The goal is to take the fewest number of samples and maintain recoverability of the source signal. First we must find an appropriate orthonormal basis $\psi_1, \ldots, \psi_N$. A signal x can be created as a linear combination of the elements ψ_i. Next, a random Gaussian matrix is created, $\Phi \in \mathbb{R}^{M \times N}$ with $M << N$. Then, a compressed vector y can be created via:

$$y = \Phi \Psi x \tag{1}$$

Where the columns of Psi are the orthonormal vectors ψ_i.

The purpose of this experiment is to demonstrate the recoverability of a compressed image. The image will be compressed using equation 1 then recovered using Orthogonal Matching Pursuit.

These are the steps used in Compressive Sensing and Signal Reconstruction.

1. First, a random gaussian matrix matrix is made, $\Phi \in \mathbb{R}^{M \times 64}$. This is the measurement matrix.
2. The image is broken into 8×8 subimages. Each sub image will be processed Compressed then recovered.
3. Each sub image is transformed into frequency space using a Discrete Cosine Transform. The 64×64 DCT matrix forms Ψ.
4. The transformed 8×8 subimage is converted into a vector $x \in \mathbb{R}^{1 \times 64}$ The order uses the zig-zag pattern demonstrated in *figure* **** to bias low frequency coefficients in Fourier space.
5. The largest s elements in x are retained. The rest are set to zero. This sparse vector is denoted x_s
6. **Compressive Sensing**: Calculate $y = \Phi \cdot x_s$. ($y \in \mathbb{R}^M$)
7. Perform OMP to recover $\hat{x} \in \mathbb{R}^{64}$. The stop condition is set to $||A \cdot \hat{x} - y||_2 < 10^{-3}$
8. $\hat{x}$ is converted to an 8×8 array using the zig-zag ordering.
9. The Inverse Discrete Cosine Transform is applied to convert the recovered subimage back to image space.

In the above steps $y = \Phi\Psi x$ is the result of compressive sensing. This process is repeated for every 8×8 subimage.

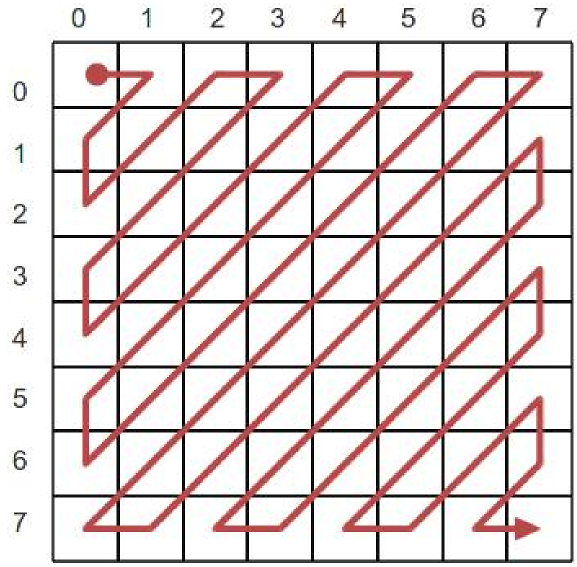

Figure 6: The Zig-zag ordering. In the DCT Basis this preferences low frequency components over higher frequency components.

Image Recovery After Compressive Sensing

This experiment was performed on four images. The imags were chosen to show a varying level of complexity. The recovered images and some recovered images are shown in figures 7-10.

Even at a very low sparsity level there is significant image reconstruction. At $s/N = 2/64$ we begin to see some texture, although finer details are not recovered. However, even at low sparsity levels, around 8, we can see more details being recovered. Notice the texture in the elephant's skin, the tree branches under the arch, and the hairs of the koala start to appear.

Figure 7: The original image of the arch (left) recovered image with s=2 (center) and recovered image with s=8 (right)

Figure 8: The original image of elephant (left) recovered image with s=2 (center) and recovered image with s=8 (right)

Figure 9: The original image of the koala (left) recovered image with s=2 (center) and recovered image with s=8 (right)

Image Recovery With Added Noise

The process of reconstructing the noisy image is the same as the previous section. Before the process of compressive sensing begins , however, random noise is added to the image. Each pixel has a random number drawn from a normal distribution. This is done twice, first with a variance of 10, then again with a variance of 50. The mean for both experiments is zero.

Variance=10

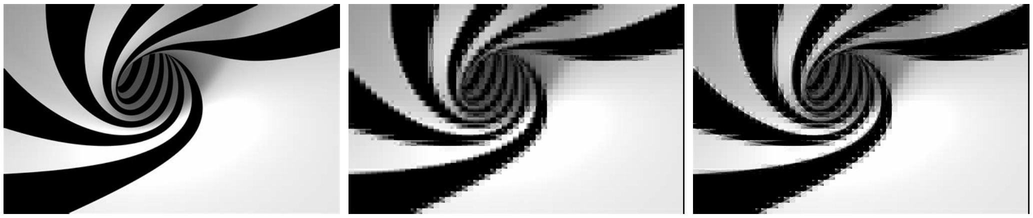

Figure 10: The original image of the spiral (left) recovered image with s=2 (center) and recovered image with s=8 (right)

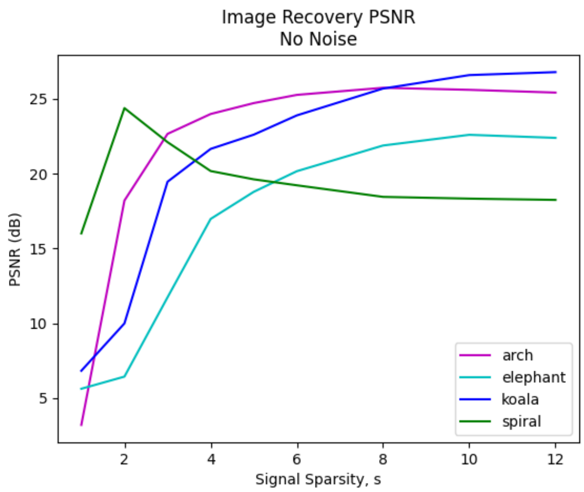

Figure 11: PSNR measured for varying sparsity levels and for different levels of added noise.

Variance=50

Figure 12: Results for image reconstruction with added noise **variance=10**. From left to right: Image with added noise. Recovery with sparsity 2. Recovery with sparsity 8.

Figure 13: Results for image reconstruction with added noise **variance=50**. From left to right: Image with added noise. Recovery with sparsity 2. Recovery with sparsity 8.

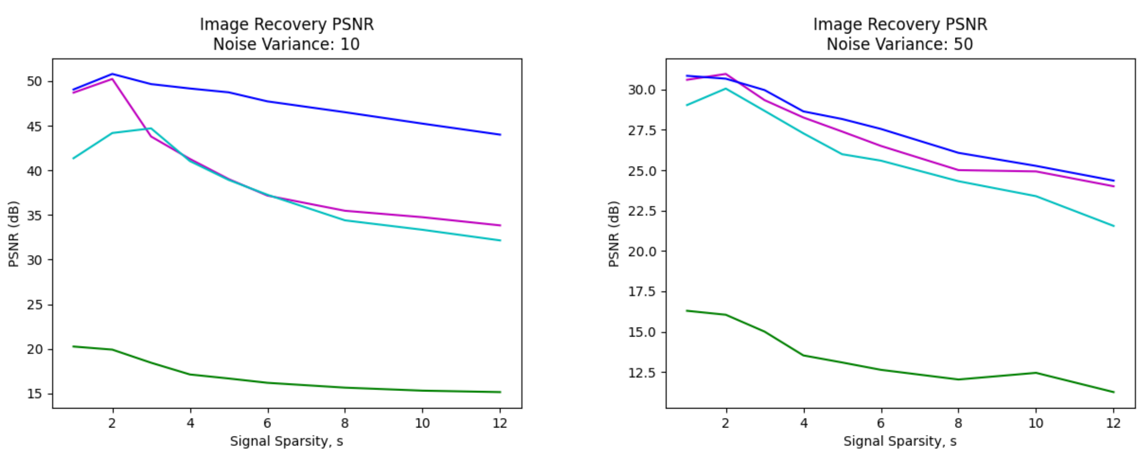

Figure 14: Peak Signal to Noise Ratio as a function of sparsity for noisy images. Top variance=10, bottom variance=50.