A review of Compressed Sensing

Rayan El Helou, Venkata Nallaparaju & Travis Taghavi

ECEN 662 – Spring 2017

1 Why is Compressed Sensing useful?

Pharr, Jakob and Humphreys [1] describe sampling as the process of taking discrete samples from continuous or non-continuous functions which could be used to reconstruct functions close to the ones we began with. They require for an ideal sampling process that the samples are equally spaced, let's say with a constant sampling period T. Subsequently, any continuous function $f(t)$ sampled at period T could be expressed as

$$f(t) = \sum_{k=-\infty}^{\infty} f(kT) s_T(t - kT)$$

where the sampling function $s_T(t)$ must satisfy the following property:

$$s_T(kT) = \begin{cases} 1 & , \quad k = 0 \\ 0 & , \quad \text{otherwise} \end{cases}, \qquad k \in \mathbb{Z}$$

and the collected samples of $f(t)$ may be represented as a vector or signal

$$\underline{f}[k] = f(kT)$$

Note that $\underline{f}$ may be a signal of length $n \in \mathbb{Z}_{>0}$ or it may be infinite in length, but for arbitrary $\underline{f}$, it is necessary to collect all its samples to perfectly reconstruct it. However, for some signals that have certain structures, we may perfectly reconstruct the signals even if we do not collect or store all its samples. Claude Shannon [2] states in his theorem that "if a function contains no frequencies higher than W cycles per second, it is completely determined by giving its ordinates at a series of points spaced $\frac{1}{2W}$ seconds apart." In other words, it is sufficient to sample at a rate more than twice the bandwidth of the signal to recover the original signal.

We are interested, in this project, to investigate how we may compress measurement of certain signals and still be able to reconstruct them, perfectly or almost perfectly. Moreover, for signals of certain structure, we may be able to sample and reconstruct the signal at a better rate than the one required by Shannon's Sampling Theorem, thus making the Nyquist-Shannon criterion a

sufficient condition, but not a necessary one. For that reason, we are interested in Compressed Sensing, described as follows.

Compressed Sensing theory, as described by Candès and Wakin [3], asks the question, "is it possible to design $m \ll n$ waveforms to capture almost all the information in $f \in \mathbb{R}^n$?" To answer such a question, Candès and Wakin introduce sensing (or measurement) of a signal as a linear transformation of the signal f as shown below.

$$y = \Phi f$$

$$f \in \mathbb{R}^n, y \in \mathbb{R}^m, m \ll n$$

The proposition is that one may reconstruct f from y even though there are much less samples in the latter than there are in the former. This can be achieved, as proposed, if the signal f is a linear transformation of a sparse signal x as follows:

$$f = \Psi x$$

$$x \in \mathbb{R}^n \therefore \Psi \in \mathbb{R}^{n \times n}$$

Hence, the choice of Φ and Ψ will dictate the quality of the signal's reconstruction.

What about sparsity in signals; what does that mean, and why is it important?

A vector is called sparse if all but a few of its entries are zero; hence, we may define an S-sparse signal as a signal with only S non-zero samples [3]. Candès and Wakin highlighted the importance of sparsity in signal reconstruction, and one key element to the success of reconstruction is the selection of Ψ.

For example, if a signal $f \in \mathbb{R}^n$ oscillates with a low number of frequencies, it can be expressed, via the Discrete Cosine Transform (DCT) as a transformation of a sparse signal x. Hence, Ψ, here, is the DCT matrix defined as follows:

$$[\Psi]_{ij} = \begin{cases} \dfrac{1}{\sqrt{n}} & , \quad i = 1 \\ \sqrt{\dfrac{2}{n}} \cos\left(\dfrac{(2j-1)(i-1)\pi}{2n}\right) & , \quad i > 1 \end{cases} \qquad , \qquad i, j = 1,2,\dots,n$$

Figure 1 below demonstrates an example of how the DCT may be used to sparsify a non-sparse signal, simply by multiplying by a square matrix, the same one defined above.

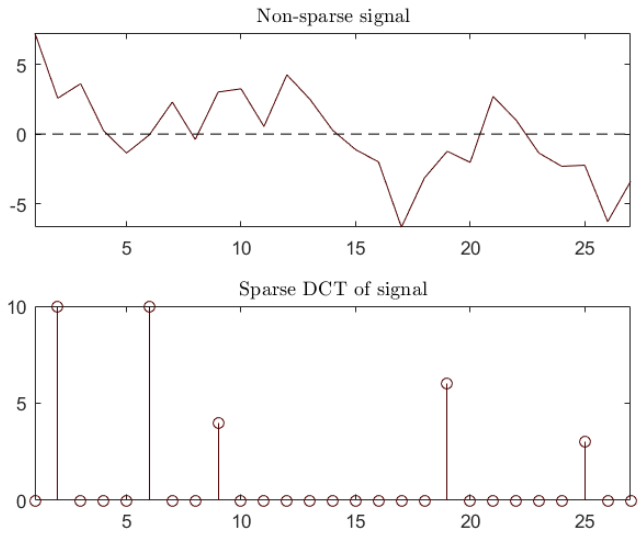

Figure 1: An example of sparsifying a signal via the DCT.

Let x_S be an S-sparse signal and $f_S = \Psi x_S$. Also let x be a signal close to x_S in the sense that $\|x - x_S\|_{l_2}$ is small, and $f = \Psi x$. Hence, if Ψ is chosen as an orthonormal basis, then the following is true:

$$\|f - f_S\|_{l_2} = \|x - x_S\|_{l_2}$$

This is a very important realization, since an 'almost S-sparse' signal and an actual S-sparse signal yield transformations that are very close to one another, which means that to reconstruct f, we must select a transformation Ψ such that x is almost sparse. Nonetheless, we have not yet introduced the actual process of reconstruction. To reconstruct f from y, the Compressed Sensing theory states that we must minimize the number of non-zero elements in x, the transform of signal f. In other words, the task is to find an estimate $\hat{x}$ such that the reconstructed signal is $\hat{f} = \Psi\hat{x}$. The estimate, given samples y, is found as follows:

$$\hat{x}(y) = \underset{\tilde{x}}{\operatorname{argmin}} \|\tilde{x}\|_0 \text{ subject to } y = \Phi\Psi\tilde{x}$$

Since $m \ll n$, there are many solutions to $y = \Phi\Psi\tilde{x}$. The idea here is that we are interested in the most sparse $\tilde{x}$, and the term $\|\tilde{x}\|_0$ is the measure for sparsity. $\|\tilde{x}\|_0$ is defined as the zero-norm of $\tilde{x}$, and one breakthrough discovery by Candès and Wakin is that the minimization of this measure is not unique to the measure itself. In fact, subject to $y = \Phi\Psi\tilde{x}$, minimizing $\|\tilde{x}\|_0$ is identical to minimizing $\|\tilde{x}\|_{l_1}$. The next section will explain this along with its relevance.

2 The reason for using the l_1-norm

Why would we be interested in minimizing the l_1-norm instead, and does it yield the same results as minimizing the zero-norm?

As a simple demonstration, we will consider a two-entry vector $z = \begin{bmatrix} x \\ y \end{bmatrix}$, where x and y are scalar. The task, as shown before is to find z with minimum zero-norm that satisfies $Az = b$. To provide a useful example, we must consider b as a scalar, our measured sample of z, to ensure that the length of b ($m = 1$) is less than that of z ($n = 2$); otherwise, it would not be a compressed sensing problem. Hence, $A \in \mathbb{R}^{1\times 2}$; for example, $A = [-m \quad 1]$. It is obvious now that the task is to find the point (x, y) that satisfies $y = mx + b$ and has minimum zero-norm, i.e. the point on a line with the least number of non-zero elements.

Remember that b is our measurement, and m is predefined and fixed. Therefore, we know certainly that the line $y = mx + b$ is either going to pass through the x-axis, the y-axis, or both. Hence, either one of the x-intercept or the y-intercept is guaranteed to ensure both 'sparsity' of vector z, since one of the elements is zero, and satisfaction of $Az = b$. Hence, such points are solutions to our problem. Strictly speaking, as previously defined, the solution to our problem will either be 1-sparse or 2-sparse (line passes through one axis or the origin respectively).

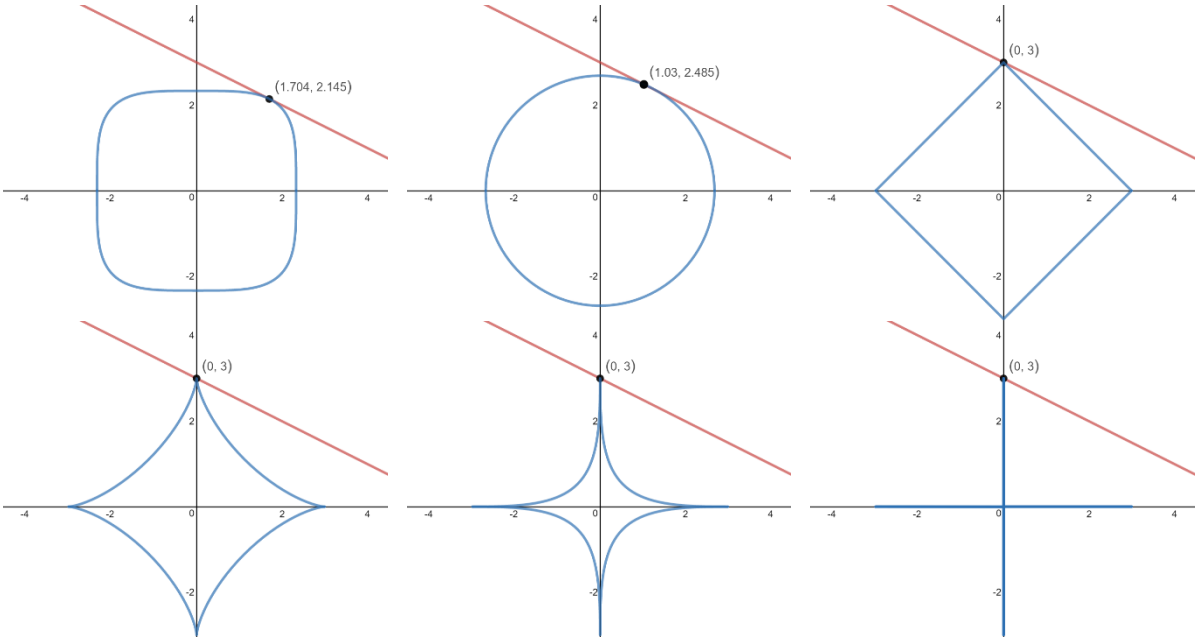

Figure 2: Minimizing the p-norm given a constraint, (top-left to right, then bottom-left to right) $p = 4, 2, 1, 0.7, 0.4, 0.1$.

As shown in the figure above, we conducted an experiment, where the variable was p, and the fixed constants were $A = [\frac{1}{2} \quad 1]$ and $b = 3$. Our objective was to find the point on $Az = b$, or $y = -\frac{1}{2}x + 3$, with minimum p-norm. Here, the p-norm of $z = \begin{bmatrix} x \\ y \end{bmatrix}$ is defined as follows:

$$\|z\|_p = (|x|^p + |y|^p)^{\frac{1}{p}}$$

We varied p from 0.1 to 4 and obtained, for each p, a shape representing the set of points with equal p-norm. By sufficiently varying the value of the p-norm, we find the single point of intersection between the shape, or isogram, and the line. Thus, we found $\hat{z}$ which minimizes the

p-norm for each p. If we were to minimize the zero-norm, defined as the number of non-zero entries, constrained to the line, then we must obtain a point, as discussed before, that lies on either the x-axis or the y-axis. Our results, some of which are shown in Figure 2, tell us that for all $0 < p \leq 1$, the solution to the minimization of the p-norm yields a point on the y-axis, the exact same solution to the zero-norm minimization. Moreover, for all $p > 1$, the minimization of the p-norm yields a non-sparse point, relatively far from the solution to the zero-norm minimization.

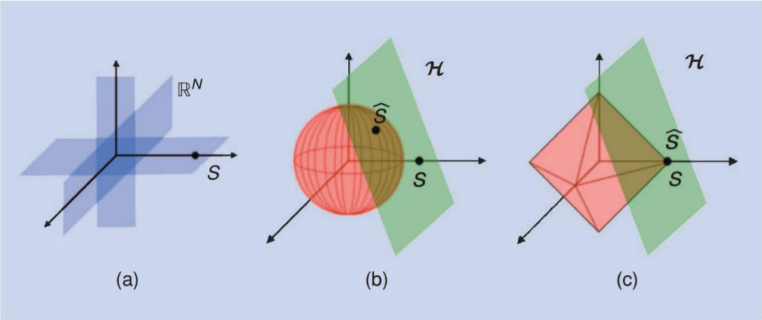

Figure 3: (a) The subspaces containing two sparse vectors in $\mathbb{R}^3$ lie close to the coordinate axes. (b) Set of points with fixed l_2-norm (red), and constraint set, $Ax = b$, (green). (c) Set of points with fixed l_1-norm (red), and constraint set, $Ax = b$, (green).

Figure 3 above, obtained from [4], demonstrates, again, that the solution to minimizing the zero-norm in three-dimensional space constraint a system of linear equations yields the same result as minimizing the 1-norm, but not the 2-norm. In the figure, the estimate using the p-norm ($p =$ 1,2) is shown as $\hat{S}$ and the actual solution, using the zero-norm, is shown as S. $S = \hat{S}$ for the 1-norm, but not for the 2-norm. Subfigure (a) shows the set of all possible solutions S for variable systems of equations. The idea is that for any A and b, as previously defined, $\underset{z}{\operatorname{argmin}}\|z\|_0 = \underset{z}{\operatorname{argmin}}\|z\|_{l_1}$ subject to the same constraint, $Az = b$.

This theory can be extended to signals of higher dimensions, indefinitely, and its idea remains the same: The solution to the zero-norm minimization problem subject to a linear transformation constraint is the vector in multi-dimensional space which is spanned by a basis of low dimensionality, i.e. contains many zero elements (i.e. sparse). Moreover, the solution to the p-norm minimization problem, as previously defined, yields the exact same solution as that to the zero-norm minimization problem, subject to the same linear constraint, for all $0 < p \leq 1$. Why is this result significant?

First, we shall introduce Linear Programming (LP), a useful tool for finding an argument that maximizes a linear function subject to linear inequalities. As defined in [5], the objective of LP is to minimize a linear function under the constraint of a set of linear inequalities. Moreover, the estimate, or solution to LP is constrained as containing non-negative elements. Mathematically, we are finding $\hat{x}$ as a function of A, b, c as follows

$$\hat{x} = \underset{\tilde{x}}{\operatorname{argmax}}\, c^T \tilde{x} \text{ subject to } Ax \leq b \text{ and } x \geq 0$$

There are many LP tools available that make it easy to solve such a problem, and it may be observed directly that one may use LP to either maximize or minimize a linear function, just by

slightly modifying the problem described above. Hence by iterating through LP and varying the constraints, one may solve the problem of l_1-norm minimization, since this norm can be expressed as a linear function of non-zero entries. For this reason, it is easier to minimize the l_1-norm than to minimize the zero-norm.

Becker, Bobin, and Candès, in [6], describe NESTA, a fast and robust first-order method which they created, that solves basis-pursuit problems. Their algorithm solves the following equation:

$$\hat{x}(b) = \underset{\tilde{x}}{\operatorname{argmin}} \|\tilde{x}\|_{l_1} \text{ subject to } \|b - A\tilde{x}\|_{l_2} \leq \epsilon$$

Thus, for $\epsilon = 0$, as they allowed, we may use their algorithm to solve our problem, reiterated as follows, by defining $b \coloneqq y$ and $A \coloneqq \Phi\Psi$.

$$\hat{x}(y) = \underset{\tilde{x}}{\operatorname{argmin}} \|\tilde{x}\|_{l_1} \text{ subject to } y = \Phi\Psi\tilde{x}$$

$$\therefore \hat{f}(y) = \Psi\hat{x}(y)$$

3 A case study, and reconstruction error analytics

By using the NESTA algorithm, which takes $A = \Phi\Psi$ and $b = y$ as inputs, we conducted an experiment by generating a set of coefficients x of length $n = 1000$ which are 15-sparse. From this sparse signal, we obtained the 'time'-domain signal $f = \Psi x$, where Ψ is the DCT matrix, as previously defined. After generating the original signal and its sparse DCT, we observe only $m \ll n$ samples from f and store them as y. Hence $y = \Phi f$, where $\Phi \in \mathbb{R}^{m \times n}$ is a full rank matrix.

Figure 4 shows the scenario where only 10% of the signal was sampled ($m = 100 \ll 1000$). You may observe that the signal, even though it's non-sparse, it was almost perfectly reconstructed (error is not observable by the naked eye).

This experiment was re-run for 100 iterations, keeping m constant. For this constant m, at each iteration, the mean-squared error between the original signal f and the reconstructed signal $\hat{f}$ was found as

$$MSE = \left\| f - \hat{f} \right\|_{l_2}$$

For the 100 iterations, the mean of this MSE was taken and stored corresponding to the constant m. This experiment was repeated for values of m ranging from 1 to 26 with increments of 0.1. As a result, we obtained means of MSE for each m, and plotted the log of these values versus the ratio of m to n as a percentage to investigate the effect of m on the reconstruction quality. The results are shown in Figure 5. The dashed line labelled 'Theoretical' is an indicator function that separates those experimental values above their mean from those below it. From these results, there seem to be a cutoff at about 15% which would suggest that one could expect to achieve close to perfect reconstruction if sampling at an m above 15% of n.

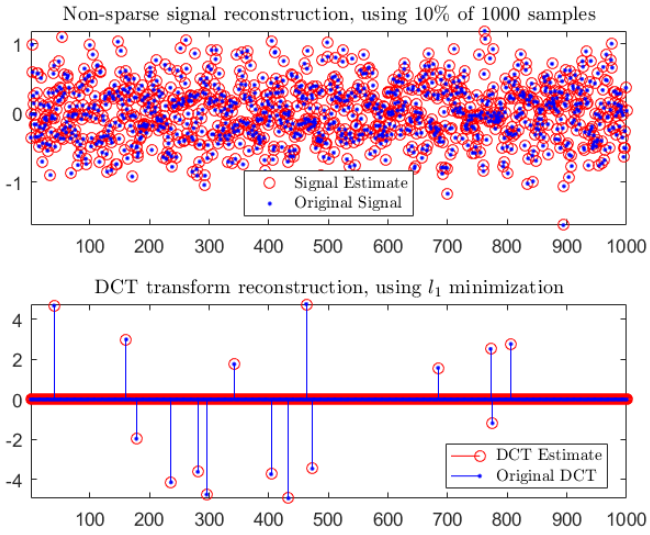

Figure 4: Signal reconstruction of signal with 15-sparse DCT.

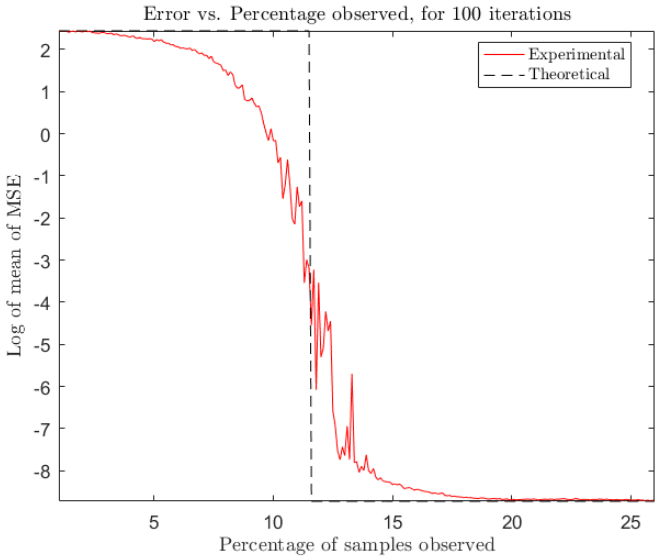

Figure 5: Error as a function of samples observed.

However, when the entire experiment is repeated with addition of White Gaussian Noise, of variable SNR, to the observation y, we observe from the results shown in Figure 6 that the same 15% cutoff appears, after which the error does not change much. This tells us that even though we would not be able to achieve close to perfect reconstruction, it's pointless to sample above the cutoff. Observe also, as the SNR value drops, the error converges to independence of m.

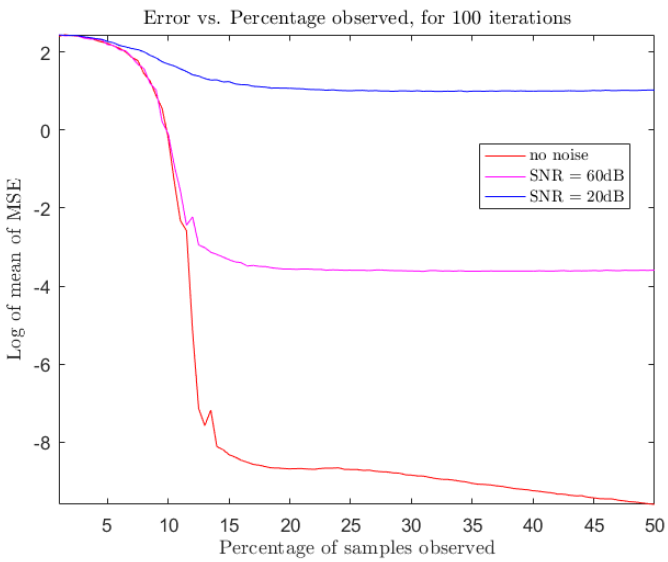

Figure 6: Error with and without noise.

This case study has been conducted using MATLAB® R2016b.

4 More examples and case studies

Please refer to the Jupyter Notebooks uploaded by our group for more examples and case studies, including image reconstruction.

5 References

[1] M. Pharr, W. Jakob and G. Humphreys. "SAMPLING AND RECONSTRUCTION." Physically based rendering. 3rd ed. N.p.: Morgan Kaufmann Publisher, 2016. Web.

[2] C. E. Shannon, “Communication in the Presence of Noise,” Proceedings of Institute of Radio Engineers, Vol. 37, No. 1, January 1949, pp. 10-21.

[3] E. Candès and M. Wakin, "An Introduction To Compressive Sampling", IEEE Signal Processing Magazine, 2008.

[4] R. G. Baraniuk, Compressive Sensing [Lecture Notes], Signal Processing Magazine, July 2007. c 2007 IEEE.)

[5] Vanderbei, R. (2007). Linear Programming: Foundations and Extensions. 3rd ed. Springer.

[6] S. Becker, J. Bobin and E. Candès, "NESTA: A Fast and Accurate First-Order Method for Sparse Recovery", SIAM Journal on Imaging Sciences, vol. 4, no. 1, pp. 1-39, 2011.