CS-754 Advanced Image Processing
Project Report
Reweighted L1-Minimization

Sanchit Jindal 200020120

Contents

I try to understand, implement and verify the claims made in the paper **Enhancing Sparsity by Reweighted 1 Minimization** by *Emmanuel J. Candès, Michael B. Wakin, Stephen P. Boyd*
A link to the paper `https://web.stanford.edu/~boyd/papers/pdf/rwl1.pdf`
The paper was published in the *Journal of Fourier Analysis and applications*

1 Brief Overview

Part of abstract from the paper:

It is now well understood that

1. It is possible to reconstruct sparse signals exactly from what appear to be highly incomplete sets of linear measurements
2. That this can be done by constrained l_1 minimization.

In this paper, we study a novel method for sparse signal recovery that in many situations outperforms l_1 minimization in the sense that substantially fewer measurements are needed for exact recovery

2 Algorithm Presented

Given

1. A vector $y \in R^n$ which is the vector that is observed
2. Sensing Matrix $A \in R^{m \times n}$

Algorithm

- Set iteration count l to zero and the weights $w_i^{(0)} = 1, \forall i = 1, 2, \ldots, n$
- solve the weighted l_1 minimization problem

$$x^{(l)} = \arg\min_x ||W^{(l)}x||_{l_1} \text{ subject to } y = \Phi x$$

  - $W^{(l)}$ is the matrix $diag(w_1^{(l)}, w_2^{(l)}, \ldots, w_n^{(l)})$

- Update the weights; for each $i = 1, 2, \ldots, n$

$$w_i^{(l+1)} = \frac{1}{|x_i^{(l)}| + \epsilon}$$

  - ϵ is added to prevent division by zero error,
    ϵ should be set slightly smaller than the expected nonzero magnitudes of x_0. In general, the recovery process tends to be reasonably robust to the choice of ϵ

- Terminate on convergence or when l attains a specified maximum number of iterations l_{max}. Otherwise, increment l and go to step 2.

3 Working

- The l_1 minimization algorithm can with better accuracy estimate the larger values of the vector

- As the weights are inversely proportional to the values, The weights of the larger, more accurate terms are decreased and the weights of the smaller terms increase,

- So in the next iteration of the algorithm more weight is given to the smaller terms and thus there estimation becomes better

4 My Implementation

To solve the Weighted Problem

$$x^{(l)} = \arg\min_x ||W^{(l)}x||_{l_1} \text{ subject to } y = \Phi x$$

- Consider $b = W^{(l)}x$

- As W is a diagonal matrix with non-zero entries on the diagonal, It will have an inverse

$$x = W^{-1}B$$

- Substituting the values in the problem we get

$$b^* = \arg\min_b ||b||_{l_1} \text{ subject to } y = \Psi b$$

  - Where Ψ is ΦW^{-1}

- Then $x = W^{-1}b^*$

5 1D Signals

5.1 Reconstruction

Code is provided in **reconstructed1D**

- Using the l1_ls library to solving for the l_1 minimization problem
- The code produces a sparse signal using parameters given at the top of the code that can be tweaked to change the input, The length of vector x, The number of spikes(*sparsity*), the Number of measurements, and the length of the spike (variance of the random number)
- A sparse random vector x, and a random orthogonal sensing matrix are generated
- The vector y is calculated using the vector x and the sensing matrix
- The function *weighted_l1* is called with different iterations number of iterations
- A figure is generated with the original image, unweighted reconstruction and weighted reconstruction with one and two iterations

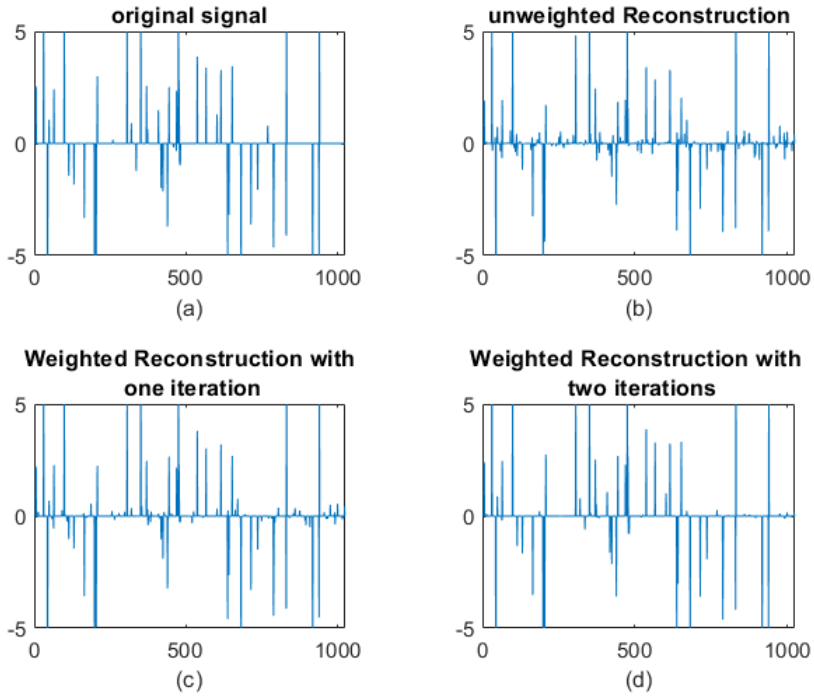

Figure 1: (reconstruction1D) Figure showing various reconstructions (a) The original vector x with length 1024, sparsity 50, number of measurements 190 (b) The Unweighted Reconstruction (c) Weighted reconstruction with one weighted iteration (d) Weighted reconstruction with 2 weighted iterations

- In (b) We can see that there are a number of non-zero elements instead of zero elements
- in (c) The "noise" reduces considerably and values of spikes also become more similar to the original
- in (d) The noise is almost gone and the signal is similar to the original
- The program also gives a relation between the RMSE error of the reconstructions

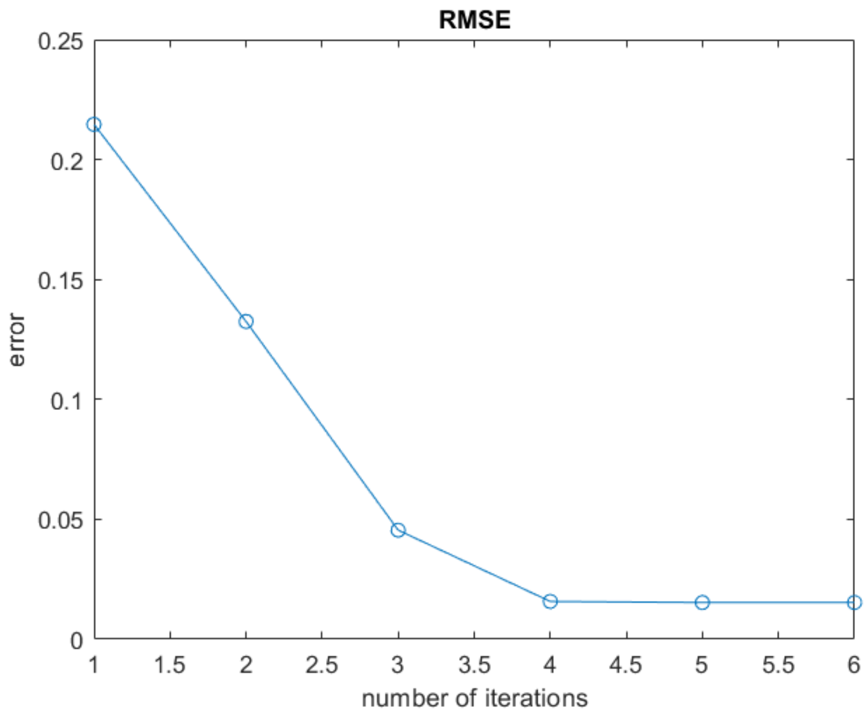

Figure 2: (error1D) The RMSE decreases sharply with increase in iterations

5.2 Relation

Code provided in **relation1D.m**

- This code also generates a sparse vector x and the sensing matrix A
- The code produces A scatter plot showing the relationship between the reconstructions and the original vector x

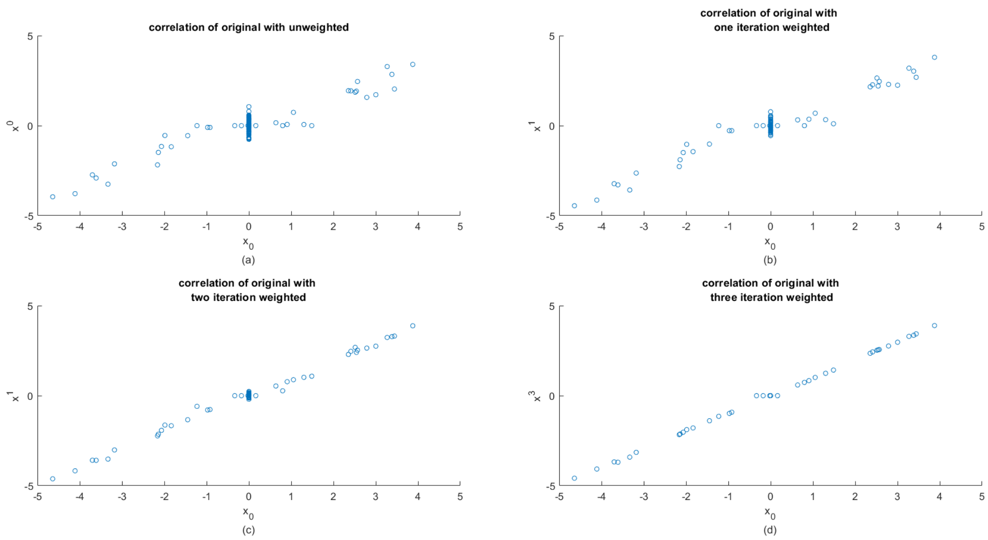

Figure 3: (relation1D) Figure showing various correlations between the elements of the original vector and the reconstruction (a) The original vector x with the unweighted reconstruction (b) Weighted reconstruction with one weighted iteration (c) Weighted reconstruction with 2 weighted iterations (d) Weighted reconstruction with 3 weighted iterations

- In (a) a lot of elements with zero values have small values in the reconstruction
- In (b),(c) The weights of the bigger values decreases and the smaller values become more important
- In (d) The relation becomes almost a linear one

6 2D Images

Reconstruction of Images using Weighted l_1 minimization
Code Provided in **Reconstruction2D.m**

6.1 Barbara Image

- Using the Barbara image for the reconstruction purposes
- Create a DCT Matrix of 64*64 as image will be sparse in DCT Basis

- Creating A patch based reconstruction using both a unweighted minimization and using re-weighted Minimization techniques
- The sensing matrix will be $\Phi * DCT$

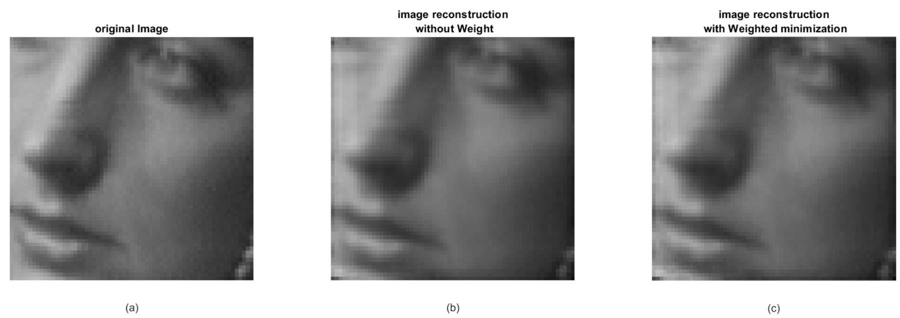

Figure 4: (Barbara.png) Figure showing reconstruction with weighted mean and with unweighted reconstruction of 64 * 64 image (a) The Original Image (b) unweighted reconstruction (c) Weighted reconstruction with 3 weighted iterations

- Using only the middle 64*64 pixels as the operation is exponential in number of bits
- in (a) the original Image is shown
- in (b) The unweighted Reconstruction is shown
- in (c) The weighted Reconstruction is shown
- We can see especially around the eyes that the weighted reconstruction is far better than the unweighted one

Figure 5: (Barbara.png) Figure showing reconstruction with weighted mean and with unweighted reconstruction of full image but stepsize is 2 (a) The Original Image (b) unweighted reconstruction (c) Weighted reconstruction with 3 weighted iterations

7 Epsilon Relation

The relation of Epsilon with the number of Iterations of the weighted Minimization The code provided in **epsilonRelation.m**

- The code visualizes the relation between the error for different values of epsilon with the number of iterations of the weighted l_1 minimization

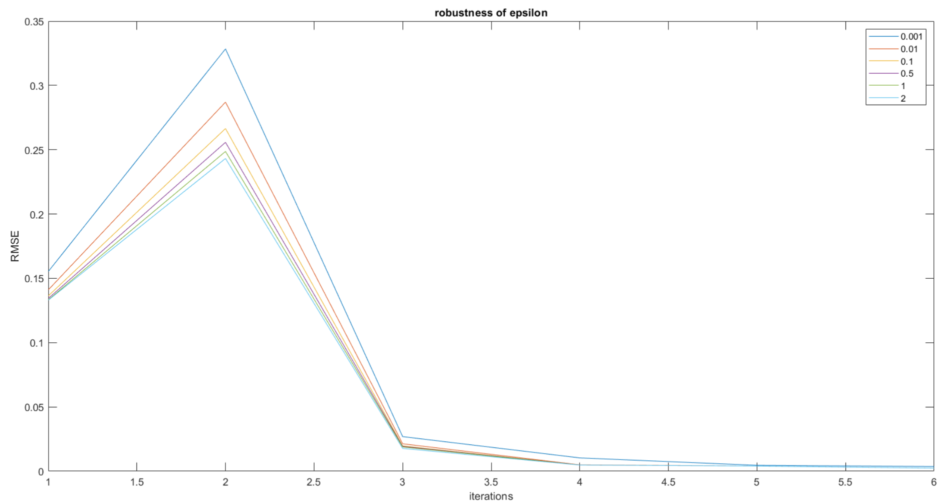

Figure 6: (epsilonRelation.png) comparison between different values of epsilon

- We can see that even though there is some difference due to the value of epsilon, They follow the same pattern
- The value of epsilon does not matter if the iterations are taken till convergence as the error decreases substantially for all

8 Noisy Measurements

Code is provided in **NoiseSignalReconstruction.m**

- The Reweighted l_1 minimization is very adapt in reconstructing a signal from a noisy Measurements
- The parameters on top of code can be changed to generate the vector x and add the noise level
- Create reconstructions for the first three iterations for the signal

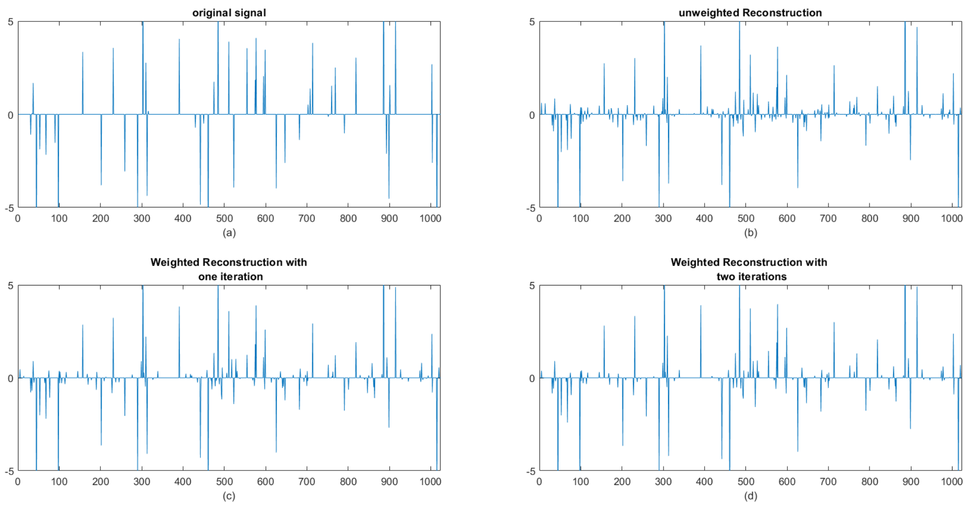

Figure 7: (noiseReconstruction.png) Reconstruction of noisy signal (a) The original signal (b) Weighted reconstruction with one weighted iteration (c) Weighted reconstruction with 2 weighted iterations (d) Weighted reconstruction with 3 weighted iterations

- It is visible in the reconstructions that it gets better with each iteration
- The noise in the signal reduces and we get more and more similar value to the original vector
- The code also produces the relation of the first 6 iteration Root Mean Square values

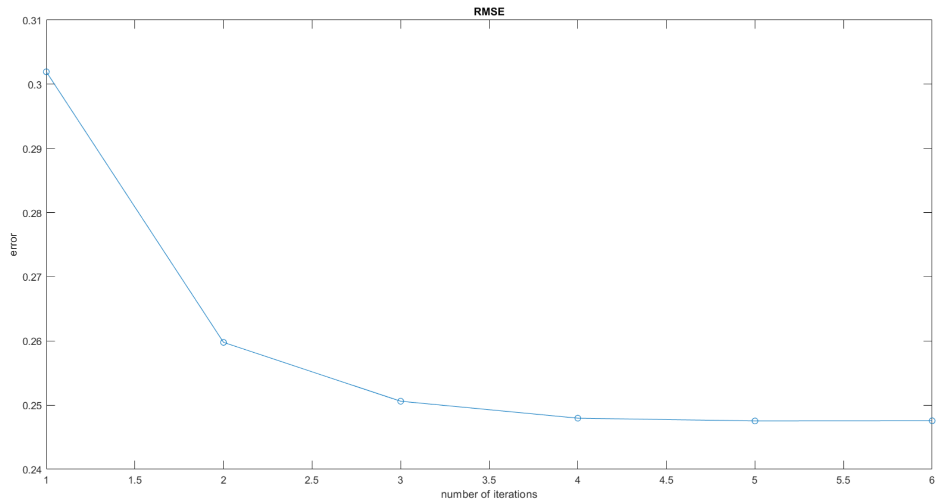

Figure 8: (noiseError.png) The RMSE decreases sharply with increase in iterations even for noisy signal

- The error is more than that of the reconstruction without error, which is obvious as with noise the reconstruction will be worse
- If the Noise Level is higher than a certain amount then the program fails With each iteration and goes further and further away from the correct value till it reaches a convergence

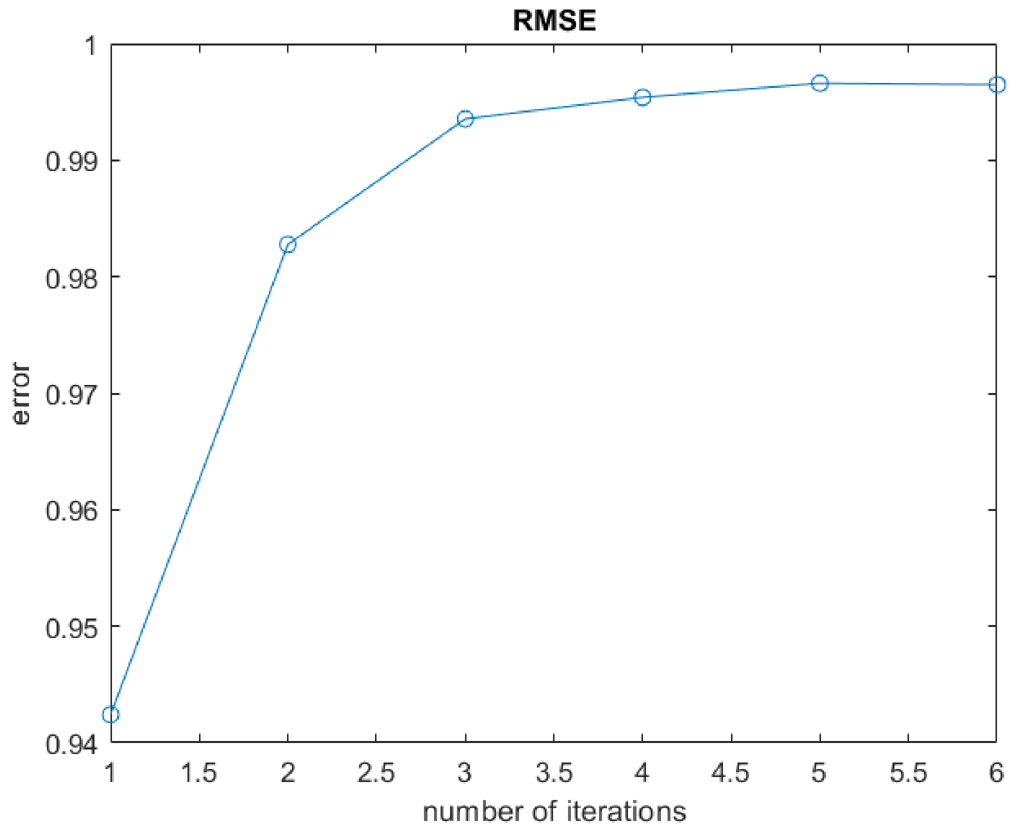

Figure 9: (highNoise.png) The RMSE increases with increase in iterations for highly noisy signal

9 Further Study

- Finding the number of iterations it takes for the program to converge? What factors does this number depend on given the measurement vector and the sensing matrix
- A method for highly noisy signals to give better results
- Though value of ϵ does not affect the convergence, finding the value of ϵ That gives the convergence in the fewest iterations
- Is there a need for randomness in the weights so that the problem always reduces to the minimum and not get stuck in a local minima.