Final Project Report: CS 754, Advanced Image Processing

Compressed Sensing With Prior Information

Adway Girish (180070002)
Fathima Zarin Faizal (180070018)

Submitted on: May 4, 2022

1 Problem Statement

Given a data vector $\boldsymbol{x} \in \mathbb{R}^N$ and a known measurement matrix $A \in \mathbb{R}^{M \times N}$, the observation vector $\boldsymbol{y} \in \mathbb{R}^M$ is obtained as $\boldsymbol{y} = A\boldsymbol{x} + \boldsymbol{n}$. As in [1], we are assuming prior information on the sparsity pattern of $\boldsymbol{x}$ (and not on the values). Informally, each entry of $\boldsymbol{x}$ has a specific probability of being non-zero that is known at the decoder.

We assume that the elements of A are drawn from a $\mathcal{N}(0, \frac{1}{M})$ distribution and that the elements of $\boldsymbol{n}$ are drawn from a $\mathcal{N}(0, \sigma^2)$ distribution. The i^{th} entry of $\boldsymbol{x}$ is given by $x_i = g_i s_i$ where $g_i \in \mathbb{R}$ is drawn according to $\mathcal{N}(0, 1)$ and s_i takes a binary value in $\{0, 1\}$. The probability of s_i being 1 is given by p_i (called the support probability) and we assume that s_i are independent. The task of the decoder is to estimate $\boldsymbol{y}$ from $\boldsymbol{x}$ when $\boldsymbol{p}$ is known to it.

The N elements of $\boldsymbol{x}$ are divided into G groups. For $n \in \{1, \ldots, G\}$, the support probabilities of all the elements in this group n (which constitute a fraction f_n of the elements of $\boldsymbol{x}$) are equal to p'_n. The number of coefficients in group n is $N_n = f_n N$.

The paper [1] proposes modifications to Basis Pursuit (BP) [2], Least Absolute Shrinkage and Selection Operator (LASSO) [3] and Orthogonal Matching Pursuit (OMP) [4] to solve this problem. We use their algorithm to obtain similar results and also propose slight modifications to their algorithms that produce better results. The performance metric that we use is p_{rec}, the fraction of terms that are correctly identified to belong to the support of $\boldsymbol{x}$, given by

$$p_{\text{rec}} = \frac{1}{S} \sum_{i=1}^{S} \frac{\left| \mathcal{I}_i \cap \widehat{\mathcal{I}}_i \right|}{|\mathcal{I}_i|},$$

where S is the number of runs per simulation, $\mathcal{I}_i$ is the true support on the i^{th} run and $\widehat{\mathcal{I}}_i$ is the support estimate on the i^{th} run.

2 Method

1. To implement BP, we first reformulate it as a linear program (LP), then use MATLAB's inbuilt linear solver to obtain the solution. This reformulation is done as follows:

$$\begin{aligned} &\text{argmin}_{\boldsymbol{x}} \ ||\boldsymbol{x}||_1 \text{ subject to } \boldsymbol{y} = A\boldsymbol{x} \\ \iff &\text{argmin}_{\boldsymbol{t}} \ \mathbf{1}^\intercal \boldsymbol{t} \text{ subject to } |x_i| \leq t_i \ \forall i \text{ and } \boldsymbol{y} = A\boldsymbol{x} \\ \iff &\text{argmin}_{\boldsymbol{t}} \ [\mathbf{0} \ \mathbf{1}]^\intercal \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{t} \end{bmatrix} \text{ subject to } \boldsymbol{y} = A\boldsymbol{x}, \ [\mathbb{I} \ \ -\mathbb{I}]^\intercal \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{t} \end{bmatrix} \leq 0 \text{ and } [-\mathbb{I} \ \ -\mathbb{I}]^\intercal \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{t} \end{bmatrix} \leq 0, \end{aligned}$$

   which is the required LP formulation, with the notation that $\boldsymbol{v} \leq 0$ for a vector $\boldsymbol{v}$ implies that $v_i \leq 0 \ \forall i$.

2. To implement LASSO, we use the Iterative Shrinkage and Thresholding Algorithm (ISTA) [5] just as discussed in class.

3. For OMP, we assume that the number of non-zero elements in $\boldsymbol{x}$, K is known. (It is possible to perform the same algorithm with comparable performance even without this information, by instead having an estimate of the number of non-zero elements as the expectation over the support probabilities, $\overline{K} = \sum_{i=1}^{G} p'_n N_n$.) The iterative procedure in OMP is repeated only K times, i.e. till the reconstructed vector is exactly K-sparse, unlike the version covered in class, where the exit condition is that the norm of the residual be lesser than some threshold.

3 Experiments and Results

We compute and plot the variation of the fraction of recovered support entries, p_{rec} as a function of M. The plots obtained are shown in Figs. 1, 2, 3. The aim is to compare the Standard BP, LASSO and OMP algorithms (represented by $+$ in the plots) with the variants that make use of prior information. For BP and LASSO, the paper [1] proposes a method where we consider a factor of $-\log p_i$ at x_i to account for the prior information (shown by $\circ$ in the plots). In addition, we propose a factor of $\frac{1}{p_i}$ (shown by $*$) and $\frac{1}{p_i} - 1$ (shown by $\triangledown$). For OMP, the paper proposes (with justification) an additive factor proportional to $\log \frac{p_i}{1-p_i}$ (shown by $+$), called Logit-Weighted OMP (LW-OMP). In each case, we average over 100 iterations.

We use the same method of generating the input signal as in the paper. We partition the $N = 240$ entries of $\boldsymbol{x}$ into four different groups as shown in Table 1. This is done such that the expected number of non-zero entries in each group is 4, i.e. $p'_n N_n = 4$, and hence the total expected number of non-zero entries, $\overline{K} = 16$.

	N_1	N_2	N_3	N_4
Simulation 1	60	60	60	60
Simulation 2	120	80	20	20
Simulation 3	204	12	12	12
Simulation 4	210	20	5	5

Table 1: The partition of N into different groups

Simulation 1 has an equal number in each group, which means that $p'_1 = \cdots = p'_4 = \frac{4}{60}$ – this is shown in red in the plots. Simulation 4, meanwhile, is at the other extreme with 210 out of the 240 entries having a tiny support probability $p'_1 = \frac{4}{210}$, while $5 + 5 = 10$ entries have a large support probability of $p'_3 = p'_4 = \frac{4}{5}$. This is shown in cyan in the plots. We also have two other sets, Simulation 2 (blue) and Simulation 3 (green), which offers a smooth transition between the extremes. The expectation is that among the four simulations, we get the best result for Simulation 4 and the worst for Simulation 1, since Simulation 4 has the heaviest influence of support probability.

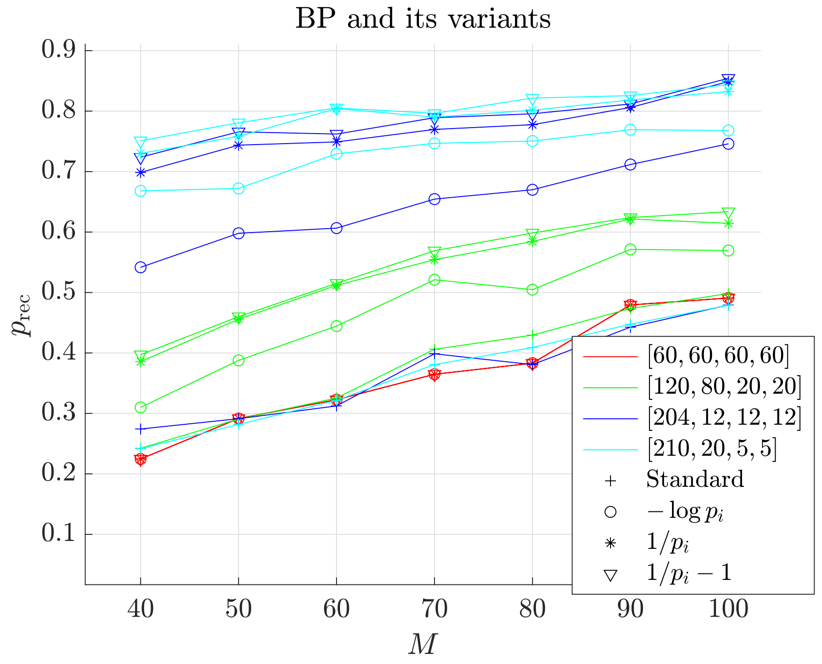

Figure 1: The variants of BP

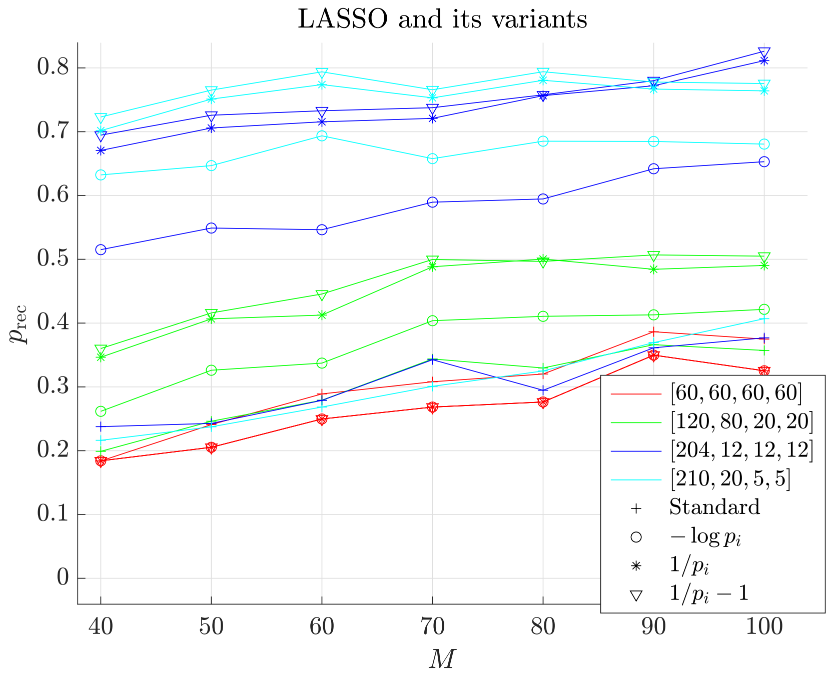

Figure 2: The variants of LASSO

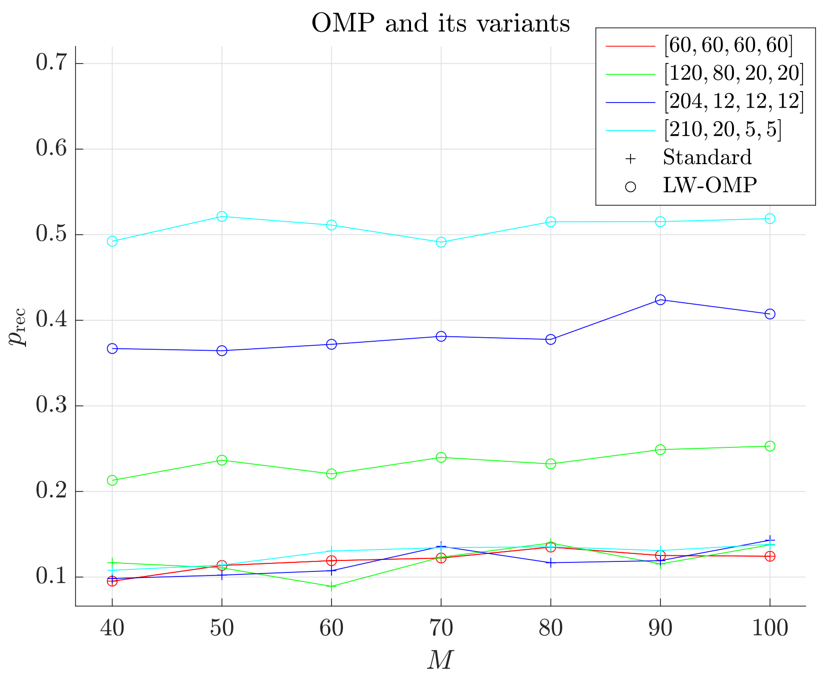

Figure 3: The variants of OMP

4 Conclusions

From the plots obtained, we make the following conclusions:

1. As expected, the performance of Simulation 1 is the worst and Simulation 4 is the best.
2. The improvement over the standard is very clear, with the percentage recovery going up from around 0.2 to over 0.7 in some cases, thus making the importance of prior information clear.
3. Unexpectedly, our chosen weights of $\frac{1}{p_i}$ and $\frac{1}{p_i} - 1$ do better than the proposed $-\log p_i$. Our conjecture is that this is due to the larger penalties that $\frac{1}{p_i}$ and $\frac{1}{p_i} - 1$ have over $-\log p_i$ (as can be seen in Fig. 4), thereby giving those entries a higher weightage.
4. Among these two, $\frac{1}{p_i} - 1$ does slightly better, possibly because of the zero penalty when $p_i = 1$.

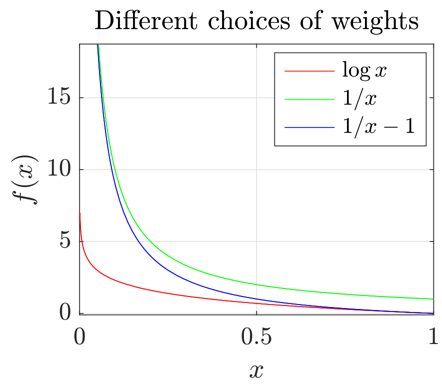

Figure 4: Plot showing the different possible weight functions

References

[1] Jonathan Scarlett, Jamie S. Evans, and Subhrakanti Dey. "Compressed Sensing With Prior Information: Information-Theoretic Limits and Practical Decoders". In: *IEEE Transactions on Signal Processing* 61.2 (Jan. 2013). Conference Name: IEEE Transactions on Signal Processing, pp. 427–439. ISSN: 1941-0476. DOI: `10.1109/TSP.2012.2225051`.

[2] E.J. Candes and T. Tao. "Decoding by linear programming". In: *IEEE Transactions on Information Theory* 51.12 (2005), pp. 4203–4215. DOI: `10.1109/TIT.2005.858979`.

[3] Robert Tibshirani. "Regression Shrinkage and Selection via the Lasso". In: *Journal of the Royal Statistical Society. Series B (Methodological)* 58.1 (1996), pp. 267–288. ISSN: 00359246. URL: `http://www.jstor.org/stable/2346178`.

[4] Y.C. Pati, R. Rezaiifar, and P.S. Krishnaprasad. "Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition". In: *Proceedings of 27th Asilomar Conference on Signals, Systems and Computers*. 1993, 40–44 vol.1. DOI: `10.1109/ACSSC.1993.342465`.

[5] Ivan W Selesnick. "SPARSE SIGNAL RESTORATION". In: (), p. 13.