Published in final edited form as:

Conf Rec Asilomar Conf Signals Syst Comput. 2016 November ; 2016: 389–393. doi:10.1109/ACSSC.

SPARSE BAYESIAN LEARNING BOOSTED BY PARTIAL ERRONEOUS SUPPORT KNOWLEDGE

Mohammad Shekaramiz, Todd K. Moon, and Jacob H. Gunther

ECE Department and Information Dynamics Laboratory, Utah State University

Abstract

Recovery of sparse signals with unknown clustering pattern in the case of having partial erroneous prior knowledge on the supports of the signal is considered. In this case, we provide a modified sparse Bayesian learning model to incorporate prior knowledge and simultaneously learn the unknown clustering pattern. For this purpose, we add one more layer to support-aided sparse Bayesian learning algorithm (SA-SBL). This layer adds a prior on the shape parameters of Gamma distributions, those modeled to account for the precision of the solution elements. We make the shape parameters depend on the total variations on the estimated supports of the solution. Based on the simulation results, we show that the proposed algorithm is able to modify its erroneous prior knowledge on the supports of the solution and learn the clustering pattern of the true signal by filtering out the incorrect supports from the estimated support set.

Keywords

Sparse Bayesian learning (SBL); single measurement vector (SMV); compressive sensing; clustered pattern; erroneous support aided

1. INTRODUCTION

Compressive sensing (CS) provides tools to represent a sparse or compressible signal from a small set of non-adaptive linear measurements [1]. In linear CS, the high dimensional signal $\mathbf{x} \in \mathbb{R}^N$ is modeled by the linear equation $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}$, where $\mathbf{A} \in \mathbb{R}^{M \times N}$ is a wide sensing matrix with $M \ll N$. The case where the sensing matrix is known has been referred to as single measurement vector (SMV) problem [2]. In the CS context, it is assumed that $\mathbf{x}$ is sparse (has few non-zero elements) under some proper basis. Besides the sparsity, in some practical applications the nonzero entries of the sparse signal $\mathbf{x}$ may appear in clusters. This feature has been referred to as clustered-pattern or block-sparsity in the literature [3, 4]. Moreover, there exist cases where a partial erroneous support set of the solution is available as a prior knowledge. This type of information may be obtained from either of the two cases below. It can be the estimate of the supports inferred from the set of measurements taken from a phenomenon of interest at the last time instant. For example, in magnetoencephalography (MEG) for the brain activities and direction of arrival (DOA) estimation problems, it turns out that when taking successive measurements, the supports of the underlying signal remain almost constant or may experience very small variations [5, 6]. This problem has been treated as a multiple measurement vector (MMV) problem with slowly time-varying supports (sources) in the literature [7, 8].

In this paper we investigate the sparse recovery problem of signals with unknown clustering pattern for the case where some prior information on the supports of the solution is available. More specifically, we assume that we are provided with a partial erroneous support set of the solution. One application of our proposed algorithm is that instead of solving an MMV with slowly time-varying supports directly, one can break the problem into a collection of SMVs and solve each SMV one at a time where they are now boosted by the support set estimated from the solution of the SMV of the previous time instant. As the second case, prior knowledge can be the output of some other CS recovery algorithm with the goal of performing some post processing to improve the overall performance in both increasing the success rate in support recovery and removing the number of indexes that have been incorrectly considered as the active supports of the solution. In both of the above cases, the available estimate of the support set usually contains a subset of the true supports accompanied with some other indexes that are incorrectly assumed to be active or they belonged to the estimated support set inferred from the last time instant set of measurements. This set of supports has been referred to as partial erroneous support set in the literature [9].

In case of having prior knowledge on the supports of the solution, some algorithms such as MBPDN and SA-SBL have been recently proposed [9,10]. MBPDN algorithm is a modified version of basis pursuit de-noising algorithm for the case where a subset of true support set is available [10]. Recently, it has been shown [9] that MBPDN is sensitive to the accuracy of the prior knowledge on the available support set. In [9], Fang et al. proposed a modified version of the conventional sparse Bayesian learning model for the purpose of using prior information on the support set in order to obtain better estimate of the true underlying sparse signal. The conventional SBL algorithm considers a Gaussian-inverse-Gamma distribution on the elements of the solution vector $\mathbf{x}$. In [9], one more layer was added to the conventional SBL. This layer incorporates a prior on the rate parameter of the Gamma distribution to take advantage of the available support knowledge of the solution. Following the same notations as was used in [9], suppose that $\mathcal{T}$ is the set of all the true supports in the solution and $\mathcal{S} \subset \mathcal{T}$ is a subset of true supports that is available. Furthermore, assume that $\mathcal{E} \subset \mathcal{T}^c$ contains the error subset that is incorrectly considered as a part of available true supports. Notice that $\mathcal{T} \cup \mathcal{T}^c = \{1, 2, \dots, N\}$ and $\mathcal{P} = \mathcal{S} \cup \mathcal{E}$, where $\mathcal{P}$ is the erroneous support set that is available to us. As discussed earlier, one can think of $\mathcal{P}$ as the support set of the previous column of the solution matrix in the MMV problem.

Below, we briefly describe the priors that were considered in [9]. Each element x_n of the solution was assumed to be drawn i.i.d. from zero mean Gaussian distribution with corresponding precision α_n , i.e.,

$$x_n \sim \mathcal{N}(x_n; 0, \alpha_n^{-1}), \quad \forall n=1, \dots, N, \quad (1)$$

where the precisions are random variables defined as

$$\alpha_n \sim \text{Gamma}(\alpha_n; a, b_n), \forall n=1, \dots, N, \quad (2)$$

where the shape parameter is fixed to $a = 10^{-10}$. In order to incorporate the available and probably erroneous support knowledge, Fang et al. [9] defined the two following cases on the rate parameters of the Gamma distributions

$$\begin{cases} b_n \sim \text{Gamma}(p, q), & \text{if } n \in \mathcal{P} \\ b_n = 10^{-15}, & \text{otherwise} \end{cases} \quad (3)$$

This means that only when the index n belongs to the set $\mathcal{P}$ the corresponding precision α_n will be governed by another Gamma distribution with hyper-parameters p and q . In this case, SA-SBL algorithm was proposed in [9].

Our proposed algorithm is essentially a modified version of SA-SBL, in which we also account for the unknown clustering pattern that may exist in the original signal. For this purpose, we incorporate the measure of clumpiness over the supports of the solution (Σ) proposed in [11] into SA-SBL algorithm. We refer to the proposed algorithm as CSA-SBL where the letter “C” stands for the “clustered” pattern signals. The main difference between our proposed algorithm and SA-SBL is that we further put a prior on the shape parameter of the Gamma distribution defined in (2) while it was set to a constant in SA-SBL [9]. Specifically, we impose that the shape parameter is to be controlled by the estimated measure of contiguity in the supports of the solution i.e., total variation on the supports of the solution. Based on the simulations, we show that this modification improves the overall performance in estimating the supports of the solution.

2. CSA-SBL ALGORITHM FOR SOLVING SMVS

In this section we describe our modified version of the conventional SBL algorithm to solve for $\mathbf{x}$ in the SMV problem defined by $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}$. It is assumed that an erroneous support set $\mathcal{P} = \mathcal{S} \cup \mathcal{E}$ is available, where $\mathcal{S}$ is a subset of the true supports of the solution and $\mathcal{E}$ is a set of incorrectly considered supports. The partition $\mathcal{S}$ and $\mathcal{E}$ in $\mathcal{P}$ is assumed unknown [9]. In order to account for the clustering pattern that may exist in the solution, we borrow the measure of clumpiness from [11], which is defined as

$$(\sum \Delta)_{(\text{support of } \mathbf{x})} = \sum_{n=2}^N |b(x_n, T) - b(x_{n-1}, T)|, \quad (4)$$

where T is a predetermined threshold. The function $b(\cdot, \cdot)$ in (4) returns a binary value and is defined as follows

$$b(x_n, T) = \begin{cases} 1 & \text{if } |x_n| > T \\ 0 & \text{otherwise.} \end{cases} \quad (5)$$

The more clustered the solution becomes, the lower value (Σ) in (4) will possess. Based on (5), the entries of $\mathbf{x}$ with the amplitude less than the threshold T are deemed to zero and their corresponding index will not be considered as supports of the solution. This is due to the fact that the elements do not have significant contribution in our measurements. We experimentally set $T = 10^{-6}$.

Below we describe the prior distributions that we consider in our hierarchical Bayesian model. Similar to SBL and for the purpose of promoting sparsity in the solution, we assume that the elements of the solution are drawn i.i.d. from a zero-mean Gaussian distribution as follows

$$x_n \sim \mathcal{N}(x_n; 0, \alpha_n^{-1}), \quad \forall n=1, \dots, N, \quad (\text{Revisiting (1)})$$

where the precision α_n is modeled as

$$\alpha_n \sim \text{Gamma}(\alpha_n; a_n, b_n), \quad \forall n=1, \dots, N. \quad (6)$$

Unlike (2), we do not assign the same shape parameter to the precisions $\alpha_n, \forall n=1, \dots, N$ in our model.

In order to incorporate the available erroneous support knowledge, we use the same model as defined [9] for the rate parameters defined in (6).

$$\begin{cases} b_n \sim \text{Gamma}(b_n; p, q), & \text{if } n \in \mathcal{P} \\ b_n = 10^{-15}, & \text{otherwise} \end{cases}, \quad (\text{Revisiting (3)})$$

where $p = q = 0.1$ as suggested in [9].

The reason for having different shape parameters for the precisions α_n in (6) is to promote the clustered pattern in the support set of the solution, where the pattern is learned via measure of total variation on the supports defined in (4). In other words, we let each shape parameter a_n be controlled via the estimated (Σ) . For this purpose, we add another hyper-prior to our model as follows

$$a_n \sim \text{Gamma}(a_n; g_n, h), \quad \forall n=1, \dots, N, \quad (7)$$

where

$$g_n := \theta \exp \{ ((\sum \Delta) - (\sum \Delta)_{n,0}) \}, \quad (8)$$

where $(\sum)$ is the initial measure of the clumpiness based on the available erroneous support set $\mathcal{P}$ and is computed from (4), and $(\sum)_{n,0}$ is the measure of clumpiness when forcing $x_n = 0$. In (8), θ is an emphasizing parameter on the amount of clumpiness over the supports of the solution and one can make it depend on the ratio M/N . This means that when the number of measurements is very low, we may not wish to emphasize on the clustered solutions. Otherwise, the algorithm may also remove some of the true supports in the set $\mathcal{P}$ due to the small number of measurements and the lack of information on the full true support set.

Remark

In the simulations, we set $h = 1$ but in general, “ h ” is selected based on the belief on the maximum permissible amplitude of the elements of $\mathbf{x}$.

Finally, the prior on the noise is defined as

$$\sigma^2 \sim \mathcal{N}(0, \gamma^{-1}), \gamma \sim \text{Gamma}(c, d), \quad (9)$$

where we set $c = d = 10^{-4}$ as suggested in [9].

According to the above prior distributions, the joint probability distribution of our model becomes

$$P(\mathbf{y}, \mathbf{x}, \boldsymbol{\alpha}, \mathbf{a}, \mathbf{b}, \sigma^2, \gamma) \propto P(\mathbf{y}|\mathbf{x}, \sigma^2 I_M) P(\mathbf{x}; 0, \boldsymbol{\alpha}^{-1}) \times P(\boldsymbol{\alpha}; \mathbf{a}, \mathbf{b}) P(\mathbf{a}; \mathbf{g}, h) p(\sigma^2; 0, \gamma^{-1}) p(\gamma; c, d) \prod_{n \in \mathcal{P}} P(b_n; p, q), \quad (10)$$

where the measurement noise is assumed to be $\mathbf{e} \sim \mathcal{N}(\mathbf{0}, \sigma^2 I_M)$.

According to (10), the marginalized posterior distributions for the variables of interest can be represented as follows. In these descriptions, conditioning on $-$, as in $(x_n|-)$, denotes the inference on x_n conditioning upon all relevant variables and the observations.

- $P(\mathbf{x}|-) \propto P(\mathbf{y}|\mathbf{x}, \sigma^2 I_M) P(\mathbf{x}|\boldsymbol{\alpha}^{-1})$. Therefore,

$$(\mathbf{x}|-) \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{x}|-}, \sum_{\mathbf{x}|-}), \quad (11)$$

where

$$\boldsymbol{\mu}_{x|-} = \gamma \sum_{x|-} A^T \mathbf{y}, \quad \sum_{x|-} = (\gamma A^T A + D)^{-1},$$

where D is a diagonal matrix with $\boldsymbol{\alpha}$ as its main diagonal i.e., $[D]_{n,n} = \alpha_n$.

- $P(\boldsymbol{\alpha}|-) \propto P(\mathbf{x}|\boldsymbol{\alpha}^{-1})P(\boldsymbol{\alpha}; \mathbf{a}, \mathbf{b})$. Therefore,

$$(\alpha_n|-) \sim \begin{cases} \text{Gamma}(\bar{a}_n, b_n + \frac{\mu_{x_n|-}^2 + \sigma_{x_n|-}^2}{2}), & \text{if } n \in \mathcal{P} \\ \text{Gamma}(\bar{a}_n, 10^{-15} + \frac{\mu_{x_n|-}^2 + \sigma_{x_n|-}^2}{2}), & \text{otherwise} \end{cases}, \quad (12)$$

In the above equation, $\bar{a}_n := a_n + \frac{1}{2}$ and $\sigma_{x_n|-}^2 := \alpha_n^{-1}$.

- $P(a_n|-) \propto P(\boldsymbol{\alpha}_n; a_n, b_n)P(a_n|g_n, h), \forall n = 1, \dots, N$. Hence

$$(a_n|-) \sim \text{Gamma}(g_n, h + \log \alpha_n), \quad \forall n = 1, \dots, N, \quad (13)$$

where $g_n = \theta \exp \{(\widehat{\sum \Delta})^{[k]} - (\widehat{\sum \Delta})_{n,0}^{[k]}\}$ and k denotes the k th iteration and $(\widehat{\sum \Delta})$ is the estimated measure of clumpiness.

- $P(b_n|-) \propto P(\boldsymbol{\alpha}_n; a_n, b_n)P(b_n; p, q), \forall n \in \mathcal{P}$. As a result,

$$(b_n|-) \sim \text{Gamma}(p, q + \alpha_n), \quad \forall n \in \mathcal{P}. \quad (14)$$

- Finally, $P(\gamma|-) \propto P(\mathbf{y}|\mathbf{x}, \gamma)P(\gamma; c, d)$ and therefore,

$$(\gamma|-) \sim \text{Gamma} \left(c + \frac{M}{2}, d + \frac{1}{2} (\|\mathbf{y} - A\boldsymbol{\mu}_{x|-}\|_2^2 + \text{tr}(A^T A \sum_{x|-})) \right). \quad (15)$$

3. SIMULATION RESULTS

In this section, we demonstrate the performance of our algorithm compared to SA-SBL algorithm proposed in [9]. For simulation purposes, the supports of the solution are randomly drawn from a Bernoulli distribution in such a way to exhibit a random clustered-sparsity pattern. In all the simulations, the number of true supports is set to $|\mathcal{T}| = 25$. The non-zero elements of solution vector $\mathbf{x}$, corresponding to true supports, are drawn i.i.d. from Gaussian distribution with zero-mean and variance $\sigma_x^2 = 1$. The sensing matrix A is $M \times 100$ and is randomly drawn from $\mathcal{N}(0, 1)$ and then it is normalized with respect to its columns. We vary the number of measurements M from 5 up to N to evaluate the performance. The elements of the noise component are drawn i.i.d. from $e_n \sim \mathcal{N}(0, \sigma_n^2)$. In all the simulations, we set SNR = 25 dB. Finally, the measurement vector $\mathbf{y}$ is computed from $\mathbf{y} = A\mathbf{x} + \mathbf{e}$. For

each run we randomly select 80% of the true supports and collect them in the set $\mathcal{S}$. Then, 10 more indexes that are not in the true supports are randomly chosen and collected into $\mathcal{E}$. We then feed both our algorithm and SA-SBL with an erroneous support set $\mathcal{P} = \{\mathcal{S} \cup \mathcal{E}\}$. Neither of the algorithms is aware of the number of true supports in set $\mathcal{P}$. For each value of M , we run 500 random cases and then average over all of the obtained results both for our algorithm and SA-SBL.

In Fig. 1, we compare the detection rate (P_D) in the support recovery between our algorithm and SA-SBL.

It can be seen in Fig. 1 that when the ratio $M/N = 0$, the detection rate $P_D = 0.8$ and it is due to the fact that we provided both algorithms with 80% of the true supports. SA-SBL algorithm provides better performance compared to CSA-SBL algorithm and the reason is it does not wish to remove its prior knowledge. It worth noting that statistically speaking, CSA-SBL provides better performance in cases of having less prior knowledge on the supports.

In Fig. 2, we compare the false alarm rate (P_{FA}) in the support estimation between our algorithm and SA-SBL. According to the results shown in Fig. 2, we see that our algorithm has a very low false alarm rate, much lower than the one obtained from SA-SBL. The performance is essentially an evidence of the role of incorporating the measure of (Σ) to account for the unknown clustering pattern in the solution. In other words, our algorithm is able to filter out those estimated supports, that cause less clustering solutions among other possible solutions.

In Fig. 3 we illustrate another measure of performance which is based on the difference between the detection and false alarm rate in the estimated support set. This figure essentially illustrates the overall performance of the algorithms based on P_D , P_{FA} , and M/N , all together. Finally, in Fig. 4, we demonstrate the performance of our modified version of SA-SBL compared to SA-SBL in terms of success rate in estimating the true clustering pattern. Our measure of success, P_r , is defined as follows

$$P_r = 1 - \frac{|(\widehat{\Sigma\Delta}) - (\Sigma\Delta)|}{(\Sigma\Delta)},$$

where (Σ) and $(\widehat{\Sigma\Delta})$ are the total number of transitions in the true support set and the estimated solution, respectively. It is clear from Fig. 4 that our propose modified version of SA-SBL (CSA-SBL) outperforms SA-SBL and that is because of the role of incorporating the measure of clumpiness in our algorithm.

Finally, we also compare the performance of SA-SBL and CSA-SBL algorithms via the following example. In this case we use the image of 112×200 pixels as illustrated in Fig. 5. In this image, we treat the black pixels as the “interesting” locations.

For the simulation purposes, we assign the value of 1 to the pixels with black color and 0 to the white ones. We then, assume that the matrix corresponding to the image is the true solution matrix $X \in \mathbf{R}^{112 \times 200}$ of an MMV problem, where the supports of the solution (sources) slowly change across the columns of X . We construct the sensing matrix in the same way we described earlier in this section. The measurements are obtained from $Y = AX + E$ with SNR=25 dB. We then break the problem into $M = 112$ SMVs, where the available information to the “ m ”th SMV is matrix A , the m th column of Y , and the estimated supports obtained from solving the SMV corresponding to the $(m - 1)$ st column of X . As illustrated in Fig. 5, the support set changes slowly across the columns of X . Therefore, even if we were provided with the true supports of the $(m - 1)$ st column of X , the information still would become the erroneous partial support set that was available for solving the “ m ”th SMV. In order to solve the SMV corresponding to the first column of X , we feed both SA-SBL and CSA-SBL with a union of 80% of the true supports of the column and 10 randomly chosen incorrect supports. In the simulations, we set the threshold to $T = 0.4$ for both SA-SBL and CSA-SBL, meaning that we discarded the estimated supports with corresponding absolute value of less than 0.4. The reconstructed images for the sampling ratio of $M/N = 0.3, 0.35, 0.4, 0.45$, and 0.5 are illustrated in Fig. 6 and Fig. 7. In order to show the quality of the reconstructed images we compare the obtained results based on peak-SNR (PSNR) as shown in Tab. 1. According to Tab. 1, we observe that CSA-SBL provides better reconstruction in terms of PSNR compared to SA-SBL algorithm.

4. CONCLUSION AND ON-GOING WORK

We proposed a hierarchical Bayesian model to solve the clustered-pattern sparse signals via single measurement vector model in case where some erroneous information about the support set of the solution is available. The proposed algorithm is essentially a modified version of the conventional sparse Bayesian learning model and SA-SBL algorithm. Based on the simulation results, we demonstrated that making the shape parameter of the Gamma distribution in the SA-SBL algorithm for the problem improves the performance in the support recovery.

References

1. Donoho D. Compressed sensing. IEEE Trans Info Th. 2006; 52:1289–1306.
2. Mishali M, Eldar YC. Reduce and boost: Recovering arbitrary sets of jointly sparse vectors. IEEE Trans Sig Proc. 2008; 56(10):4692–4702.
3. Zhang Z, Rao B. Recovery of block sparse signals using the framework of block sparse Bayesian learning. IEEE Int Conf on Acoust, Speech and Sig Proc (ICASSP). 2012:3345–3348.
4. Fang J, Shen Y, Li H, Wang P. Pattern-coupled sparse Bayesian learning for recovery of block-sparse signals. IEEE Trans Sig Proc. 2015; 63(2):360–372.
5. Kwon HE, Rao B. On the benefits of the block-sparsity structure in sparse signal recovery. IEEE Int Conf on Acoust, Speech and Sig Proc (ICASSP). 2012:3685–3688.
6. uo JL, Zhang XP, Wang Z. Direction-of-arrival estimation using sparse variable projection optimization. Int Symp on Circs and Syst (ISCAS). May.2012 :3106–3109.
7. Kim JM, Lee OK, Ye JC. Dynamic sparse support tracking with multiple measurement vectors using compressive music. IEEE Int Conf on Acoust, Speech and Sig Proc (ICASSP). 2012:2717–2720.

8. Zhang, Z., Rao, BD. Exploiting correlation in sparse signal recovery problems: Multiple measurement vectors, block sparsity, and time-varying sparsity. ICML11 Workshop Structured Sparsity: Learning and Inference; Washington, DC. 2011.
9. Fang J, Shen Y, Li F, Li H, Chen Z. Support knowledge-aided sparse Bayesian learning for compressed sensing. IEEE Int Conf on Acoust, Speech and Sig Proc (ICASSP). 2015:3786–3790.
10. Vaswani N, Lu W. Modified-CS: Modifying compressive sensing for problems with partially known support. IEEE Trans Sig Proc. 2010; 58(9):4595–4607.
11. Shekaramiz, M., Moon, T., Gunther, J. On the block-sparse solution of single measurement vectors. 49th Asilomar Conf. on Sig., Syst. and Compt; Nov. 2015; p. 508-512.

Figure 1.
Detection rate comparison.

Figure 2.
False alarm rate comparison.

Figure 3.

An overall measure of performance in terms of both detection rate and false alarm rate.

Figure 4.
Comparison in terms of pattern recovery.

Figure 5.
True Image.

Figure 6.
Results of reconstructed images for SA-SBL for different sampling ratios (M/N).

Figure 7.
Results of reconstructed images for CSA-SBL for different sampling ratios (M/N).

Table 1

NMSE and PSNR comparison in image reconstruction.

	SA-SBL	CSA-SBL
Sampling ratio (M/N)	PSNR (dB)	PSNR (dB)
$M/N = 0.30$	12.8467	13.5155
$M/N = 0.35$	15.1456	15.8384
$M/N = 0.40$	16.4610	18.1877
$M/N = 0.45$	18.6170	30.9498
$M/N = 0.50$	Inf	Inf