COMPRESSED SENSING

WEIGHTED APPROACH TO COMPRESSED SENSING WITH APPLICATIONS TO EEG SOURCE LOCALIZATION AND MIMO RADAR DOA ESTIMATION

By

AHMED AL HILLI

A dissertation submitted to the

School of Graduate Studies

Rutgers, The State University of New Jersey

In partial fulfillment of the requirements

For the degree of

Doctor of Philosophy

Graduate Program in Electrical and Computer Engineering

Written under the direction of

Athina Petropulu and Laleh Najafizadeh

And approved by

_____

_____

_____

_____

New Brunswick, New Jersey

May, 2019

ABSTRACT OF THE DISSERTATION

Compressed Sensing

Weighted approach to compressed sensing with applications to EEG source localization and MIMO radar DOA Estimation

By: AHMED AL HILLI

Dissertation Directors:

Prof. Athina Petropulu

Prof. Laleh Najafizadeh

Sparse signals can be recovered based on fewer samples than suggested by the Nyquist Theorem. Those samples are obtained during a process referred to as sparse sampling, which amounts to collecting random projections of the signal on some basis functions. Using the collected projections, and under certain conditions, the sparse signal can be estimated using a non-linear estimation process. The idea is to estimate a vector with smallest number of non-zero entries, or equivalently find the least ℓ_0-norm solution. In sparse signal recovery problems, ℓ_1-norm minimization is typically used as relaxation of the more complex ℓ_0-norm minimization problem. Conditions for strong equivalence between ℓ_0-norm and ℓ_1-norm include Mutual Coherence, Restricted Isometry Property (RIP), and Null Space Property (NSP). The Range Space Property (RSP) provides the conditions under which the least ℓ_1-norm solution is equal to at most one of the least ℓ_0-norm solutions. These conditions depend on the sensing matrix and the support of the underlying sparse solution.

The ℓ_1-norm minimization method has been applied successfully in many applications. However, ℓ_1-norm minimization method may not satisfy the RSP conditions. In this thesis, we first address the problem of recovering sparse signals which arise in scenarios that do not satisfy the RSP conditions. For such cases we propose to formulate and solve a weighted ℓ_1-norm minimization problem, in which the sensing matrix is post-multiplied by a diagonal weight matrix. We show that by appropriately choosing the weights, we can formulate an ℓ_1-norm minimization problem that satisfies the RSP, even if the original problem does not. By solving the weighted problem we can obtain the support of the original problem. We provide the conditions which the weights must satisfy, for both noise free and noisy cases. Although those conditions involve information about the support of the sparse vector, the class of good weights is very wide, and in most cases encompasses a low-resolution estimate of the underlying vector, for example, an estimate that is obtained via a simple method that does not encourage sparsity.

The proposed weighted approach is applied to the problem of the Electroencephalography (EEG) source localization, in which, the obtained measurements from sensors distributed around the head are used to localize sources inside the brain. Assuming sparse brain activity in response to simple tasks, one can formulate source localization problem as a sparse signal recovery problem, in which the support of the sparse vector is directly related to the coordinates of the sources inside the brain. However, the corresponding basis matrix, referred to as the *lead field* matrix, has high mutual coherence, and there is no guarantee that the corresponding least ℓ_1-norm solution will solve for the actual locations. Developing reliable EEG source localization techniques has potential applications in Brain Computer Interfaces (BCIs). Most of existing EEG-based BCIs rely on the scalp recorded signals, but the poor spatial resolution of EEG limits the number of actions to be discriminated. Source domain information can improve the discrimination of actions, which motivates the application of source localization in EEG-based BCIs. The proposed method, with weights equal to the Multiple Signal Classification (MUSIC) estimate of the brain activity, is used in an experiment eliciting auditory evoked potentials, and is shown to correctly localize brain activations.

The main issue with ℓ_1-norm minimization approaches is that the global minimum associated with ℓ_1-norm cost function may not coincide with the sparsest solution. The Sparse Bayesian Learning (SBL) method has been shown to have shown tighter approximation to the ℓ_0-norm function, and its global minimum for noise free case coincide with the sparsest solution. In the second part of this thesis, we propose a Weighted Sparse Bayesian Learning (WSBL). Unlike SBL, where all hyperparameter priors follow Gamma distributions with identical parameters, in WSBL, the hyperparameters are Gamma distributed with distinct parameters. These parameters, guided by some known weights, give more importance to some hyperparameters over others, thus introducing more degrees of freedom to the problem and leading to better recovery performance. The weights can be determined based on a low-resolution estimate of the sparse vector, for example an estimate obtained via a method that does not encourage sparsity. The choice of the MUSIC estimate as weight is analyzed. Unlike SBL, where the hypeparameters are not bounded, in WSBL there is an upper bound; this make it easy to select a threshold that distinguishes between zero and non-zero elements in the recovered sparse vector, which helps the iterative recovery process converge faster. Theoretical analysis based on variational approximation theory, and also simulation results demonstrate that WSBL results in substantial improvement in terms of probability of detection and probability of false alarm, as compared to SBL and support knowledge-aided sparse Bayesian (BSN), especially in the low signal to noise ratio regime. The performance of WSBL is evaluated for Direction of Arrival (DOA) in colocated Multiple Input Multiple Output (MIMO) radar.

While WSBL exhibits substantial improvement over SBL, its performance depends highly on MUSIC estimates, which may suffer when there is no adequate number of snapshots, for example, in cases in which the structure of the sparse vector changes with time. In the last part of this thesis, we propose Bernoulli Sparse Bayesian Learning (BSBL), in which, a machine learning approach is used to estimate the probability of each entry in the sparse vector to be non-zero. Unlike MUSIC, these probabilities are estimated based on a single snapshot. In BSBL, each rate parameter, b_i, is modeled as a Bernoulli random variable, taking a high or a low value with probability p_i and

$1 - p_i$, respectively. The probability p_i is estimated based on the observation, and a statistical model that describes how different rate parameters give rise to different outputs; given the sensing matrix, a specific signal-to-noise ratio level, and the degree of sparsity, the latter model can be obtained during a training phase. In particular, a Gaussian Naive Bayesian Classifier (NBC) is used to assign each b_i to the *high* or *low* value class, corresponding to active or non-active elements of the sparse vector, based on the computed probability. Based on the estimated rate parameters, BSBL estimates the hyperparameters along the lines of SBL. The proposed approach shows significant improvement in probability of detection and false alarm, as compared to SBL-type methods at low Signal to Noise Ratio (SNR) and various sparsity levels.

Acknowledgements

I would like to express my very great appreciation to my advisers Professor Athina Petropulu and Professor Laleh Najafizadeh for their constructive suggestions and kind guidance which were of great help for the completion of this thesis.

I would like to offer my special thanks to the Faculty members of ECE department at Rutgers University for their teaching effort.

Last, but not least, I would like to express my deep gratitude to my family and friends for their unlimited support during my study.

Dedication

To the souls of my Father and Mother, and to Family

Table of Contents

Chapter 1

Introduction

Sparse vector recovery problems arise in many applications including biomedical imaging [1–5], and radars [6–11]. The goal there is to find the sparse vector with the smallest number of non-zero entries that give rise to the observed vector. Mathematically, the problem is formulated as an ℓ_0-norm minimization problem. However, due to the associated complexity [12], the relaxed ℓ_1-norm minimization problem is usually solved instead. The solution of the ℓ_0-norm minimization problem coincides with that of ℓ_1-norm minimization problem if certain conditions are satisfied, including the Restricted Isometry Property (RIP) [13], the Null Space Property (NSP) [14], the Mutual Coherence [15], or the Range Space Property (RSP) [16]. In practice, however, the aforementioned conditions may not be met, in which case the least ℓ_1-norm solution is not related to the sparsest solution. Re-weighted iterative approaches have also been proposed for sparse vector estimation. In FOCal Underdetermined System Solver (FOCUSS) [17] a reweighted ℓ_2-norm minimization problem is solved, making use of initial weights. Although FOCUSS has been shown to converge, it may converge to a local minimum [17]. In [18], a reweighted iterative ℓ_1-norm algorithm for enhancing sparsity is proposed. However, no convergence guarantees are provided in [18]. For cases in which the ℓ_1-norm minimization problem does not yield the least ℓ_0-norm solution, a weighted approach was proposed in [2], which exploits available information, such as a low resolution estimate of the sparse vector, to bias the solution and obtain the underlying sparse vector. An alternative to the ℓ_1-norm minimization methods is FOCUSS [17], which minimizes the ℓ_p (with $0 \leq p < 1$) norm. Although this minimization problem tends to the ℓ_0 minimization problem as p tends to 0, FOCUSS suffers from local minima. Sparse signal recovery from overcomplete dictionaries has found many

applications in signal processing domains [19]. In such applications, we seek to find a solution that has minimum number of non-zero entries that best describe the observation. Finding such solution has been proved to be NP-hard problem [12]. Under certain conditions, like Restricted Isometry Property (RIP) [13], Null Space Property (NSP) [14], and Mutual Coherence [15], ℓ_1-norm problem is used to find such a solution.

Re-weighted recursive methods, namely, FOCUSS [20] and re-weighted ℓ_1-norm minimization [18], have also been proposed for sparse vector recovery. In FOCUSS, initial low resolution estimate is used as weights, and at each iteration, minimum least-square problem is solved. The weights are then updated according to the solution obtained in the previous iteration. It has been shown that FOCUSS iteration converges. However, due to multiple local minima, it may converge to a sub-optimal solution [19]. Also, FOCUSS performance degrades in low SNR scenario [21, 22]. The re-weighted ℓ_1-norm approach [18] has also been proposed for enhancing sparsity of the final sparse estimation. In this approach, a sparse signal estimate is obtained by solving a weighed ℓ_1-norm minimization, and the weights are constructed according to the the previous estimation of the sparse vector. The re-weighted ℓ_1-minimization approach suffers from the local minima, and no convergence guarantee.

Greedy algorithms have also been proposed to reduce the complexity associated with the ℓ_1-norm minimization. The most commonly greedy approaches include Matching Pursuit (MP) [23], Orthogonal MP (OMP) [24], and Order Recursive MP (ORMP) [25]. In MP, the best bases that describe the observations are selected iteratively. In each iteration, the most correlated basis with the residual vector is selected, and then, the selected basis contribution is subtracted form the residual vector. The main issue with MP is that the algorithm can selected the same basis at different iterations. This issue has been addressed in OMP, where the residual is updated by projecting the observation vector onto the subspace of the selected bases. This prevent the selection of the same basis in different iterations. Another variant of OMP is ORMP. All greedy algorithms select the basis using the correlation with the residual, which may suffer in cases of correlated sources. For instance, in [25], degraded performance is observed in cases of data that consists of closely spaced frequencies; for high resolution Discrete Fourier

Transform dictionary matrix.

Another class of sparse vector recovery methods are the probabilistic approaches, which can provide a tighter approximation to the ℓ_0-norm function, and its global minimum for noise free case coincide with the sparsest solution. In these sparse vector recovery methods, the estimated sparse solution is obtained by maximizing the Bayesian posterior using sparsity inducing priors [26–28]. In Sparse Bayesian Learning (SBL), the entries of the sparse vector assumed to follow a Gaussian priors with distinct variances. These variances are referred to as hyperparameters, which follow Gamma distributions with shape parameter a and rate parameter b. These hyperparameters are estimated by maximizing the marginal likelihood function. Due to the known property of Expectation-Maximization, It was shown that SBL always converges [19]. SBL has been used for classification [26], regression [26], and sparse basis selection [19]. In [29], support knowledge-aided sparse Bayesian approach was proposed. In this approach, erroneous prior knowledge of the support is assumed to be available. The rate parameters that belong to the erroneous support are modeled as random variables, which are estimated with the hyperparameters. The rate parameters that do not belong to the erroneous support are assigned to low values. In [30], Multiple response Sparse Bayesian Learning (MSBL) was proposed as an extension of SBL for cases of multiple snapshots. In this approach, the sources are assumed to be stationary, and do no change across the observed snapshots. When noise is present, SBL-based approaches estimate the noise variance along with the hyperparameters. On the other hand, non-probabilistic methods, such as Basis Pursuit De-Noising (BPDN) [31, 32], reweighted ℓ_1-norm [18], and weighted ℓ_1-norm [2], assume the noise variance to be known in advance.

1.1 Contributions of the Dissertation

1.1.1 Weighted ℓ_1-norm Approach

For the weighted ℓ_1-norm approach, we address the problem of recovering sparse signals by weighting the corresponding sensing matrix with a diagonal matrix. We show that by appropriately choosing the weights, one can reformulate the ℓ_1-norm minimization

to a problem that satisfies RSP conditions, even if the original problem does not. Using this approach, we solve for a sparse vector with the same support as the underlying sparse vector.

For the weighted ℓ_1-norm approach, we also provide conditions for the weights such that the weighted ℓ_1-norm approach satisfies the RSP conditions. We first provide conditions based on sufficient conditions for RSP, and then we relax the conditions to include a wider class of weights.

Simulation results show that in practical scenarios in which one does not know whether the RSP is met, using the proposed approach results is significantly improved localization of the sparse signal. The proposed approach overcomes the difficulties associated with ℓ_1-norm minimization in scenarios under which the conditions of strict equivalence may not hold. Unlike the weighting method in [20] the proposed approach has less sensitivity to noise, and it is less sensitive than greedy algorithms [23–25] in scenarios with correlated sources.

The proposed method, with weights equal to the MUSIC estimate, is used in an EEG experiment eliciting auditory evoked potentials to localize the brain activity. The results show that the proposed approach can identify activities in the proximity of the expected brain regions.

The results of this work have been published in

- A. Al Hilli, L. Najafizadeh and A. Petropulu, "EEG sparse source localization via Range Space Rotation," IEEE 6^{th} International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), Cancun, 2015, pp. 265-268.

- A. A. Hilli, L. Najafizadeh and A. Petropulu, "Sparse target scene reconstruction for SAR using range space rotation," IEEE Radar Conference (RadarConf), Philadelphia, PA, 2016, pp. 1-5.

- A. Al Hilli, L. Najafizadeh and A. P. Petropulu, "A Weighted Approach for Sparse Signal Support Estimation with Application to EEG Source Localization," in IEEE Transactions on Signal Processing, vol. 65, no. 24, pp. 6551-6565, Dec.,

2017.

1.1.2 Weighted Sparse Bayesian Learning

For the sparse Bayesian approach, we propose Weighted SBL (WSBL) for sparse signal recovery. Unlike SBL, where all hyperparameter priors follow the same distributions with the same parameters, in WSBL, the hyper-parameters have a distinct set of parameters. This introduces more degrees of freedom of this optimization and thus improves accuracy. These parameters act as weights, giving more importance to some hyperparameters over others. The relative importance of the hyperparameters can be determined based on a low-resolution estimate of the sparse vector, for example an estimate obtained via a method that does not encourage sparsity.

Theoretical analysis based on variational approximation theory, and also simulation results demonstrate that WSBL results in substantial improvement in terms of probability of detection and probability of false alarm, as compared to SBL, especially in the low signal to noise ratio regime.

An important advantage of WSBL over SBL is that it makes it easy to select the threshold that distinguishes between zero and non-zero elements in the recovered sparse vector, which reduces the convergence time of WSBL as compared to SBL. Both WSBL and SBL have the same complexity in the beginning of the iteration, and their complexity reduces as more columns are excluded from the estimation. However, in WSBL, due to the upper limit on the hyperparameters, the number of excluded columns is larger. This results in faster convergence time for WSBL as compared to SBL, and as a result, reduced complexity. We have validated theoretically the use of MUSIC as weights with WSBL.

The proposed approach is a good candidate for target estimation in Multiple Input Multiple Output (MIMO) radar, a technology that has received significant attention recently. MIMO radars can achieve high resolution with fewer antennas as compared to conventional radars. MIMO radar with millimeter wave technology is currently being evaluated for applications in Advanced Driver Assistance Systems (ADAS), like Automatic Emergency Breaking (AEB). With mm-wave technology, the range cell is

small, thus, it is reasonable to expect that there is only a small number of targets in the range cell, thus the target scene is sparse, and target estimation can be formulated as a sparse signal recovery problem. The proposed WSBL is applied to estimate DOA of targets for the colocated MIMO radar scenario, and its performance is compared to well known Bayesian approaches.

The results of this work have been published in

- A. A. Hilli, L. Najafizadeh and A. Petropulu, "Weighted sparse Bayesian learning (WSBL) for basis selection in linear underdetermined systems," 4th International Workshop on Compressed Sensing Theory and its Applications to Radar, Sonar and Remote Sensing (CoSeRa), Aachen, 2016, pp. 115-119.

- A. A. Hilli and A. Petropulu, "MIMO radar using sparse sensing: A weighted sparse Bayesian learning (WSBL) approach," 51st Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, 2017, pp. 80-84.

- A. A. Hilli, L. Najafizadeh and A. Petropulu, "Weighted Sparse Bayesian Learning (WSBL) for Basis Selection in Linear Underdetermined Systems," IEEE Transaction on Vehicular Technology, Under review.

1.1.3 Bernoulli Sparse Bayesian Learning (BSBL)

In the last part of this thesis, we propose the Bernoulli Sparse Bayesian Learning, in which a machine learning approach is deployed to increase the probability of detection rate. In Bernoulli Sparse Bayesian approach, an additional layer of random variables have been added to increase the degree of freedom, thus increasing the accuracy of sparse estimation. In this approach, the shape parameters are still assigned to low values, while rate parameters b_i is modeled as Bernoulli random variable with parameter pi; the rate parameters are to be estimated along with the hyperparameters. The Bernoulli parameters p_i are estimated based on the observation vector. For a Gamma distributed random variable, a high rate parameter makes the pdf more concentrated around zero, corresponding to zero hyperparameter with higher probability, or equivalently to anon-zero value in the corresponding entry of the sparse vector. Thus, the entry of $\mathbf{x}$ which

has high p_i belongs to the support of the final vector sparse estimate. Unlike WSBL, p_is can be estimated with low complexity.

The proposed BSBL deploys machine learning approach to estimate the Bernoulli random variable parameters. We propose to use Naive Bayes Classifier (NBC) to estimate such parameters in a computationally efficient manner.

We show that MIMO radar DOA estimation, BSBL exhibits improved probability of detection as compared to WSBL. WSBL performance degrades due to degradation in MUSIC estimate for low SNR and increased number of targets. BSBL performance depends on NBC, which can be trained to deal with low SNR cases, and increased number of sources.

The results of this work have been published in

- A. A. Hilli, and A. Petropulu, "Bernoulli Sparse Bayesian Learning for Basis Selection," The 20^{th} IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC 2019), Under review.

1.2 Notation

Throughout the thesis we use bold face capital letters to denote matrices, bold face small letters to denote column vectors, and normal small letters to denote scalars. We use $\prec$ to represent element-wise less than, $\succeq$ to represent element-wise greater than, $\preceq$ to represent element-wise greater than or equal, $(.)^{\dagger}$ to represent the Moore-Penrose pseudoinverse, $(.)^T$ to represent matrix transpose, and $N(\mathbf{A})$ to represent the null space of matrix $\mathbf{A}$. x_i is used to represent the i-th element of the vector $\mathbf{x}$, $\mathbf{a}_i$ is used to represent the i-th column of a matrix $\mathbf{A}$, and w_{ij} the (i,j) element of matrix $\mathbf{W}$. $\|.\|_p$ is used to denote the p-norm of a vector, while $|\mathbf{x}|$ denotes a vector whose elements are the absolute values of the corresponding elements of $\mathbf{x}$.

Chapter 2

A Weighted Approach for Sparse Signal Support Estimation with Application to EEG Source Localization

2.1 Introduction

In sparse signal recovery problems one wishes to describe the observation, $\mathbf{y}$, using the smallest possible basis from a dictionary matrix, $\mathbf{A}$, or equivalently, one seeks the sparsest solution to the problem $\mathbf{y} = \mathbf{A}\mathbf{x}$. In order to avoid the complexity of the underlying ℓ_0-norm minimization problem, one typically employs ℓ_1-norm minimization, hoping that the two problems are *strictly equivalent* [33], i.e., there is a unique least ℓ_0-norm solution (a unique sparsest solution), which coincides with the least ℓ_1-norm solution. Conditions for strict equivalence include the mutual coherence [34], the Restricted Isometry Property (RIP) [13], and the Null Space Property [14]. When ℓ_1-norm minimization algorithms are used for sparse signal recovery [13, 34, 35], it is implicitly assumed that the conditions for strict equivalence hold. However, in most cases, either the conditions do not hold, or their validity cannot be easily confirmed. In fact, only a few dictionary matrices have been proven to satisfy the strict equivalence conditions [13, 36]. In real world scenarios, strict equivalence conditions may not be satisfied, in which case the least ℓ_1-norm solution may not be related to the sparse signal of interest.

The recently introduced Range-Space Property (RSP) and full rank property [33] address the case in which the least ℓ_0-norm solution is not unique, and provide the conditions for the least ℓ_1-norm solution to be equal to at most one of the sparsest solutions; when this happens, the ℓ_1-norm minimization problem and the ℓ_0-norm problems are

Work supported in part by NSF under grant NSF ECCS 1408437.

called *equivalent*. The RSP conditions depend on the sensing matrix and the support of the underlying sparse solution.

Another approach for estimating a sparse vector encompasses re-weighted recursive methods, most notably, the FOcal Underdetermined System Solver (FOCUSS) method [20], and the re-weighted ℓ_1-norm minimization [18]. In FOCUSS, a low resolution estimate of the sparse vector is used as an initial weight. In an iterative fashion, FOCUSS solves a weighted minimum least-squares problem, using weights that are proportional to the solution obtained during the previous iteration. Although the iteration has been shown to converge, due to multiple local minima, it may converge to a sub-optimal solution [19]. Also, FOCUSS performance degrades at low Signal-to-Noise Ratio (SNR) [21,22]. The re-weighted ℓ_1-norm approach [18] is also iterative; in each iteration, a sparse signal estimate is obtained by solving a weighted ℓ_1-norm minimization problem subject to various constraints, with the weights taken to be inversely proportional to the absolute values of the signal estimate obtained in the previous iteration. In addition to problems with local minima, there is no guarantee that the re-weighted ℓ_1-norm approach will converge as the number of iterations increases. Conditions for uniqueness of the weighted ℓ_0-norm problem are discussed in [37], where a recursive approach to estimate the unique least ℓ_0-norm solution is proposed.

Greedy algorithms have also been proposed for sparse vector recovery, such as the Matching Pursuit (MP) [23], the Orthogonal MP (OMP) [24], and the Order Recursive MP (ORMP) [25]. The MP method constructs a basis that best represents the signal by selecting columns out of the overcomplete basis matrix, $\mathbf{A}$. The basis is constructed one vector at a time. In the i-th iteration, the column that is most correlated with the residual vector is selected and added to the basis; the residual vector is initially set equal to the observation, and in each iteration is updated by subtracting from it the contribution of the selected column in that iteration. In OMP [24], the estimate of the sparse vector is updated by projecting the observation vector onto the subspace of the selected columns. This avoids the selection of the same column in different iterations, which can happen in MP. A variant of OMP is the Order Recursive Matching Pursuit (ORMP) [23]. Since in all greedy algorithms column selection depends on

the column correlation with the residual, the performance suffers in cases of highly correlated sources. For instance, in [25], the authors show degraded performance in cases of data consisting of closely spaced frequencies; and dictionary matrix constructed from the rows of a high resolution Discrete Fourier Transform.

In this chapter, we address the problem of recovering sparse signals by weighting the corresponding sensing matrix with a diagonal matrix. We show that by appropriately choosing the weights, we can formulate an ℓ_1-norm minimization problem that satisfies the RSP, even if the original problem does not. By solving the weighted problem we obtain the support of the original solution. We provide the conditions which the weights must satisfy, for both the noise free and the noisy cases. We should emphasize that like all existing weighted methods [18, 20], the proposed method does not provide a precise construction of weights. Instead, our work shows analytically that there is a large class of functions that qualify as good weights; these functions are related to a low resolution estimate of the underlying sparse vector, for example, an estimate obtained via a standard method that does not induce sparsity. It is worth mentioning that after estimating the support, one can easily recover the signal by solving an overdetermined system based on the observations and a matrix composed of the columns of the dictionary matrix associated with the support. Simulation results show that in practical scenarios in which one does not know whether the RSP is met, using the proposed approach results are significantly improved localization of the sparse signal samples. Simulation results show that the proposed method performs better than [20], and is less sensitive than greedy algorithms [23–25] in scenarios with correlated sources.

As an example of application for the proposed approach, we focus on the Electroencephalography (EEG) source localization problem, in which, measurements obtained by sensors placed on the head are used to localize activations inside the brain. Assuming sparse brain activity in response to simple tasks, source localization can be formulated as a sparse signal recovery problem, where the support of the sparse vector is directly related to the coordinates of the sources inside the brain. The challenge, however, is that the corresponding basis matrix, referred to as the *lead field* matrix, has high mutual coherence, based on which, there is no guarantee that the corresponding least

ℓ_1-norm solution will lead to the actual sources. The proposed method, with weights equal to the MUSIC estimate of the brain activity, is used in an experiment eliciting auditory evoked potentials, and is shown to correctly localize brain activations.

The chapter is organized as follows. Section 2.2 reviews background theory on sparse vector recovery. Section 2.3 presents the proposed approach. Section 2.4 shows the simulation and experimental results of the application of the proposed approach in EEG source localization, and Section 2.5 provides conclusion remarks.

2.2 Background theory on Sparse Signal Recovery

The sparsest solution of the underdetermined problem

$$\mathbf{y} = \mathbf{A}\mathbf{x} \tag{2.1}$$

can be obtained by solving an ℓ_0-norm minimization problem, i.e.,

$$\begin{aligned} \underset{\mathbf{x}}{\text{minimize}} \quad & \|\mathbf{x}\|_0 \\ \text{subject to} \quad & \mathbf{y} = \mathbf{A}\mathbf{x}. \end{aligned} \tag{2.2}$$

However, since this is an NP-hard problem, its convex ℓ_1-norm relaxation is used instead [34], [13], [35], i.e.,

$$\begin{aligned} \underset{\mathbf{x}}{\text{minimize}} \quad & \|\mathbf{x}\|_1 \\ \text{subject to} \quad & \mathbf{y} = \mathbf{A}\mathbf{x}. \end{aligned} \tag{2.3}$$

Under certain conditions, the problems (2.2) and (2.3) are strictly equivalent, i.e., there is a unique least ℓ_0-norm solution that coincides with the least ℓ_1-norm solution [20], [13, 14, 34]. Sometimes, (2.2) has several sparsest solutions, i.e., solutions with the same number of non-zero elements but different support. When one of those sparsest solutions coincides with the solution of (2.3), then, the problems of (2.3) and (2.2) are referred to as equivalent [33]. The work in [33] provides a set of conditions for the least ℓ_1-norm solution to be equal to a sparsest solution, and also shows that under those conditions there is at most one such sparsest solution that satisfies the RSP conditions. The conditions are given in the following theorem.

Theorem 1. *Let* $\mathbf{x} \in \mathbf{R}^n$ *be a sparsest solution to (2.1).* $\mathbf{x}$ *is the least* ℓ_1*-norm solution if and only if* $\mathbf{x}$ *satisfies both of the following conditions:*

(i) Range Space Property (RSP): There exists a vector $\mathbf{u}$ *such that*

$$\begin{cases} (a)\ \mathbf{u} \in R(\mathbf{A}^T) \\ (b)\ u_i = 1 \qquad x_i > 0 \\ (c)\ u_i = -1 \qquad x_i < 0 \\ (d)\ |u_i| < 1 \qquad x_i = 0 \end{cases}$$

(ii) Full Rank Property: The matrix $[\ \mathbf{A}_{J+} \quad \mathbf{A}_{J-}\ ]$ *has full column rank, where* $\mathbf{A}_{J+}$ *and* $\mathbf{A}_{J-}$ *are matrices containing the columns of* $\mathbf{A}$ *associated with the positive and negative elements of* $\mathbf{x}$*, respectively* [33].

Corollary 1 [33]: *For any given underdetermined linear system, there exists at most one sparsest solution satisfying the RSP* [33].

2.3 The proposed approach

2.3.1 Noise-Free Sparse Vectors

As it was already mentioned, if the sensing matrix $\mathbf{A}$ in (2.1) does not satisfy the strong equivalence conditions, there is no guarantee that the least ℓ_1-norm solution will be equal to the underlying sparse vector $\mathbf{x}$. In the following, we show that by appropriately weighting the sensing matrix, we can formulate an ℓ_1-norm minimization problem that satisfies the RSP, thus guaranteeing that its solution has the same support with at most one of the sparsest solutions. By selecting the weights based on an estimate obtained via a conventional method that does not encourage sparsity, we bias the ℓ_1-norm minimization problem to produce a sparsest solution that has the same support as the signal of interest.

A tool in showing the aforementioned result is a sufficient condition for Theorem 1 to hold. The sufficient condition is given in the following theorem, in the context of the system of (2.1).

Theorem 2. *If for a sparsest solution* $\mathbf{x}$ *it holds that* $\left|\mathbf{A}_{J0}^T(\mathbf{A}_{Js}^T)^\dagger \mathbf{u}_s\right| \prec \mathbf{1}$ *, then* $\mathbf{x}$

equals the least ℓ_1-norm solution, where $\mathbf{1}$ *is a vector of* 1*'s of appropriate size,* $\mathbf{A}_{Js} \triangleq [\ \mathbf{A}_{J+}\ \ \mathbf{A}_{J-}\ ]$*,* $\mathbf{A}_{J0}$ *contains the columns of* $\mathbf{A}$ *associated with zero values in* $\mathbf{x}$*, and* $\mathbf{u}_s = [\mathbf{u}_p^T\ \mathbf{u}_n^T]^T$ *where* $\mathbf{u}_p$ *and* $\mathbf{u}_n$ *are vectors of* 1*'s and* -1*'s corresponding to positive and negative* x_i*s respectively.*

Proof. See Appendix A. □

Let $\mathbf{W}$ be a positive diagonal matrix, which is nonzero over the support of $\mathbf{x}$, such that $\mathbf{x} = \mathbf{Wq}$. In the following, when we use the term "weight vector" we refer to the vector containing the diagonal elements of $\mathbf{W}$. Then (2.2) can be written as

$$\begin{aligned} \underset{\mathbf{q}}{\text{minimize}} \quad & \|\mathbf{Wq}\|_0 \\ \text{subject to} \quad & \mathbf{y} = \mathbf{AWq}. \end{aligned} \tag{2.4}$$

Since $\|\mathbf{Wq}\|_0 = \|\mathbf{q}\|_0$, we can write (2.4) as

$$\begin{aligned} \underset{\mathbf{q}}{\text{minimize}} \quad & \|\mathbf{q}\|_0 \\ \text{subject to} \quad & \mathbf{y} = \mathbf{AWq}. \end{aligned} \tag{2.5}$$

The corresponding ℓ_1-norm minimization problem becomes

$$\begin{aligned} \underset{\mathbf{q}}{\text{minimize}} \quad & \|\mathbf{q}\|_1 \\ \text{subject to} \quad & \mathbf{y} = \mathbf{AWq}. \end{aligned} \tag{2.6}$$

By solving the problem of (2.6), we will be able to determine the support of $\mathbf{x}$, as $\mathbf{x}$ and $\mathbf{q}$ have the same support. For the discussion below, we will assume that the observation is a linear combination of independent columns of $\mathbf{A}$.

Suppose that the problem of (2.3) does not satisfy the RSP. When formulating the problem of (2.6), we can appropriately select $\mathbf{W}$ such that the solution of (2.6) satisfies the condition in Theorem 2, and as such, the problem of (2.6) satisfies the RSP. With the right $\mathbf{W}$, and since based on Corollary 1 there is at most 1 sparsest solution satisfying the RSP, the solution of (2.6) would have the same support as the least ℓ_0-norm solution of (1).

Substituting $\mathbf{A}$ with $\mathbf{AW}$ into the condition of Theorem 2, we get the following sufficient condition for $\mathbf{W}$:

$$|\mathbf{W}_{J0}\mathbf{A}_{J0}^T(\mathbf{A}_{Js}^T)^\dagger\mathbf{W}_{Js}^{-1}\mathbf{u}_s| \prec \mathbf{1}, \tag{2.7}$$

where $\mathbf{W}_{J0}$ and $\mathbf{W}_{Js}$ contain elements of $\mathbf{W}$ associated with zero and non-zero elements of $\mathbf{x}$, respectively.

Proposition 1: If we choose $\mathbf{W}$ such that (2.7) is satisfied on a sparsest solution $\mathbf{q}^\star$, then $\mathbf{q}^\star$ is the only sparsest solution that is equal to the solution of (2.6).

The proof follows directly from Corollary 1.

We should emphasize that the weighting matrix does not improve the coherence of the sensing matrix, since multiplication with a diagonal matrix does not change the matrix coherence. Instead, we can constrain the weighting matrix so that the weighted problem satisfies the Range Space Property. By satisfying the RSP, the least ℓ_1-norm solution will have the same support as at most one of the sparsest solutions of the original ℓ_0-norm problem.

There are infinitely many choices for $\mathbf{W}$ that validate the condition in (2.7). In this work, we propose to use as weight vector an estimate of the sparse vector obtained by a method that does not encourage sparsity. We should note that ℓ_1-norm solution would not be a good weight vector because most of its entries are zeros, and important components of the true signal may have been lost. Suppose that a rough estimate of $\mathbf{x}$ can be approximated as $\hat{\mathbf{x}}$, where

$$\hat{x}_j = \sum_{i\in S} |x_i| h_i(j-i), \tag{2.8}$$

where $h_i(j)$ is a Gaussian kernel with zero mean and variance σ_i, and S is the set of indices in the support of the sparse vector. Let us set $w_j = \hat{x}_j$. Then, (2.7) can be written as

$$|\mathbf{W}_{J0}\mathbf{A}_{J0}^T(\mathbf{A}_{Js}^T)^\dagger\mathbf{U}\mathbf{w}_{-J}| \prec \mathbf{1}, \tag{2.9}$$

where $\mathbf{U}$ is a diagonal matrix with diagonal entries equal to $\mathbf{u}_s$, and $\mathbf{w}_{-J}$ is a column vector with entries equal to the reciprocal of diagonal entries of $\mathbf{W}_{Js}$. Based on the

inequality $|\mathbf{a}^T\mathbf{b}| \leq \|\mathbf{a}\|_\infty \|\mathbf{b}\|_1$, and taking $\mathbf{b} = \mathbf{w}_{-J}$, we can rewrite (2.9) as

$$\begin{aligned} \|(\mathbf{a}_j^T(\mathbf{A}_{Js}^T)^\dagger \mathbf{U})^T\|_\infty w_{jj} \sum_{k=1}^{K} w_{-Jk} < 1 \\ \|(\mathbf{a}_j^T(\mathbf{A}_{Js}^T)^\dagger \mathbf{U})^T\|_\infty < \frac{1}{\sum_{i\in S} w_{jj}/|w_{ii}|}, \end{aligned} \tag{2.10}$$

for all $j \notin S$, where K represents the number of non-zero entries in $\mathbf{x}$, $\mathbf{a}_j$ represents the column of the matrix $\mathbf{A} \notin S$, and w_{jj} represent the j-th diagonal element of $\mathbf{W}$.

The term $\|(\mathbf{a}_{J0j}^T(\mathbf{A}_{Js}^T)^\dagger \mathbf{U})^T\|_\infty$ of (2.10) can be written as $\|(\mathbf{a}_{J0j}^T \mathbf{A}_{Js}(\mathbf{A}_{Js}^T\mathbf{A}_{Js})^{-1}\mathbf{U})^T\|_\infty$ which is basically related to the coherence of the matrix $\mathbf{A}$. Without the weights ($\sigma_i = 0$), high coherence might prevent (2.10) from being valid. However, if $\sigma \neq 0$, there would be weights to counteract the high coherence and make (2.10) valid.

2.3.2 A More Relaxed Condition on W

While (2.7) provides a sufficient condition for $\mathbf{W}$ in order for RSP to be satisfied, there is a wider class ofweights that satisfy the RSP. Those can be found by exploiting a more relaxed condition for matrices $\mathbf{W}$ that satisfy the RSP. To provide such condition, we first rewrite the RSP conditions in an equivalent form, and then provide an upper limit for the weights that meet the RSP.

Let $\mathbf{x}_0$ be a sparsest solution to (2.1). Based on the RSP, there should be a vector $\mathbf{v}$ such that

$$\begin{aligned} &\text{(a)} \quad \mathbf{A}_{Js}^T\mathbf{v} = \mathbf{u}_s \\ &\text{(b)} \quad |\mathbf{A}_{J0}^T\mathbf{v}| \prec \mathbf{1}, \end{aligned} \tag{2.11}$$

with $\mathbf{A}_{Js}$, $\mathbf{A}_{J0}$, and $\mathbf{u}_s$ as defined in Theorem 2.

If we assume that the number of non-zero entries in the sparse vector $\mathbf{x}_0$ is less than the number of rows in $\mathbf{A}$, (2.11-(a)) is an underdetermined system, and its solution can be written as

$$\mathbf{v} = (\mathbf{A}_{Js}^T)^\dagger \mathbf{u}_s + \boldsymbol{\alpha}, \tag{2.12}$$

where $\boldsymbol{\alpha}$ is a vector that belongs to the null space of $\mathbf{A}_{Js}^T$. Substituting (2.12) into (2.11), we get

$$\begin{aligned} &\text{(a)} \quad |\mathbf{A}_{J0}^T(\mathbf{A}_{Js}^T)^\dagger \mathbf{u}_s + \mathbf{A}_{J0}^T\boldsymbol{\alpha}| \prec \mathbf{1} \\ &\text{(b)} \quad \boldsymbol{\alpha} \in N(\mathbf{A}_{Js}^T), \end{aligned} \tag{2.13}$$

Note that the conditions in (2.13) are sufficient and necessary for $\mathbf{x}_0$ to be equal to the least ℓ_1-norm solution. In other words, $\mathbf{x}_0$ equals the least ℓ_1-norm solution if and only if the intersection of the set of vectors $\boldsymbol{\alpha}$ that satisfy (2.13-(a)) with the null space of $\mathbf{A}_{Js}^T$ is a non-empty set. Finding such $\boldsymbol{\alpha}$ can be done by solving the following convex problem

$$\begin{aligned} \underset{\boldsymbol{\alpha},\epsilon}{\text{minimize}} \quad & \epsilon \\ \text{subject to} \quad & |\mathbf{A}_{J0}^T(\mathbf{A}_{Js}^T)^\dagger \mathbf{u}_s + \mathbf{A}_{J0}^T\boldsymbol{\alpha}| \preceq \mathbf{1}\epsilon \\ & \mathbf{A}_{Js}^T\boldsymbol{\alpha} = 0. \end{aligned} \tag{2.14}$$

If the minimum ϵ is such that < 1, then the conditions in (2.13) are satisfied. Otherwise, there is no $\boldsymbol{\alpha} \in N(\mathbf{A}_{Js}^T)$ that satisfies (2.13-a). For the weighted approach, the conditions of (2.13) can be rewritten as

$$\begin{aligned} &\text{(a)} \quad |\mathbf{W}_{J0}\mathbf{A}_{J0}^T(\mathbf{A}_{Js}^T)^\dagger \mathbf{W}_{Js}^{-1}\mathbf{u}_s + \mathbf{W}_{J0}\mathbf{A}_{J0}^T\boldsymbol{\alpha}| \prec \mathbf{1} \\ &\text{(b)} \quad \boldsymbol{\alpha} \in N(\mathbf{A}_{Js}^T) \end{aligned} \tag{2.15}$$

If the sufficient condition of Theorem 2 is not satisfied for the non-weighted ℓ_1-norm problem (i.e., some of the elements in the vector $\left|\mathbf{A}_{J0}^T(\mathbf{A}_{Js}^T)^\dagger \mathbf{u}_s\right|$ have values larger than or equal to 1), $\mathbf{x}_0$ can still be a solution to the ℓ_1-norm problem if there is a vector in the null space of $\mathbf{A}_{Js}^T$ that can meet (2.13-(a)). However, due to the high coherence between the columns of $\mathbf{A}$, many elements in vector $\left|\mathbf{A}_{J0}^T(\mathbf{A}_{Js}^T)^\dagger \mathbf{u}_s\right|$ may have values greater than 1, and it could be impossible to find such an $\boldsymbol{\alpha}$ (i.e., the intersection between the two sets that are defined in ((2.13)-(a) and (b)) is the empty set). However, we can ensure that the intersection of the sets defined in ((2.15)-(a) and (b)) is non-empty set by choosing $\mathbf{W}$ such that ((2.15)-(a)) is satisfied for a vector $\boldsymbol{\alpha} \in N(\mathbf{A}_{Js}^T)$. Next, we find an upper limit for $\mathbf{W}$ that follows the model in (2.8), such that the conditions in (2.15) are satisfied.

Following the model in (2.8), the problem can be recast as selecting the maximum σ such that there is an intersection between the sets of $\boldsymbol{\alpha}$s that satisfy (2.15)-a and

(2.15)-b. This can be done by solving the following optimization problem

$$\begin{aligned} \underset{\boldsymbol{\alpha},\sigma}{\text{maximize}} \quad & \sigma \\ \text{subject to} \quad & |\mathbf{W}_{J0}\mathbf{A}_{J0}^T(\mathbf{A}_{Js}^T)^\dagger\mathbf{W}_{Js}^{-1}\mathbf{u}_s + \mathbf{W}_{J0}\mathbf{A}_{J0}^T\boldsymbol{\alpha}| \prec \mathbf{1} \\ & \mathbf{A}_{Js}^T\boldsymbol{\alpha} = 0 \\ & w_{J0i} = \sum_{i\in S}|x_i|h_i(j-i),\ \ \forall j \notin S, \end{aligned} \tag{2.16}$$

where $h_i(j)$ is as defined in (2.8). We can solve (2.16) via an iterative approach as follows. We first choose a large value for σ, and find $\boldsymbol{\alpha}$ such that the distance between the two sets defined in the first and second constraints in (2.16) is minimum. Based on that $\boldsymbol{\alpha}$, we can construct the maximum weights such that the first constraint in (2.16) is satisfied. The maximum weights can be used to estimate σ such that the weights in the third constraint in (2.16) are smaller than the maximum weights. The process is repeated by finding another $\boldsymbol{\alpha}$ with minimum distance between the two set defined in the first and second constraints in (2.16) using the new weights. The iteration stops when the distance is zero.

Remark The above approach does not provide a practical way to construct the weights, as it uses information about the sparse signal support. However, by finding the limit for the weights, one can see that the class of weights that satisfy the conditions in (2.15) is wide. A low-resolution estimate of the true sparse vector would probably fall in that class, and thus make the weighted ℓ_1-norm problem satisfy the RSP, even if the original non-weighted ℓ_1-norm problem does not satisfy the RSP. This is illustrated in the following simulations.

Matrix $\mathbf{A}$ (64×2898) was constructed using a realistic head model obtained from BrainStorm software [38]; the matrix models the relationship between the EEG channel readings and the source distribution inside the brain. The number of non-zero entries in the sparse vector was set to 4. The non-zero indices were assigned randomly, and the corresponding non-zero elements were drawn from a Gaussian distribution with mean 1 and standard deviation 5. In order to simulate a weighting matrix obtained based on a blurred version of the sparse vector, the diagonal weighting matrix was constructed according to (2.8) with $\sigma_i = \sigma$, for $i = 1, ..., 4$, and we considered $(\sigma = 1, 2, .., 7)$ and

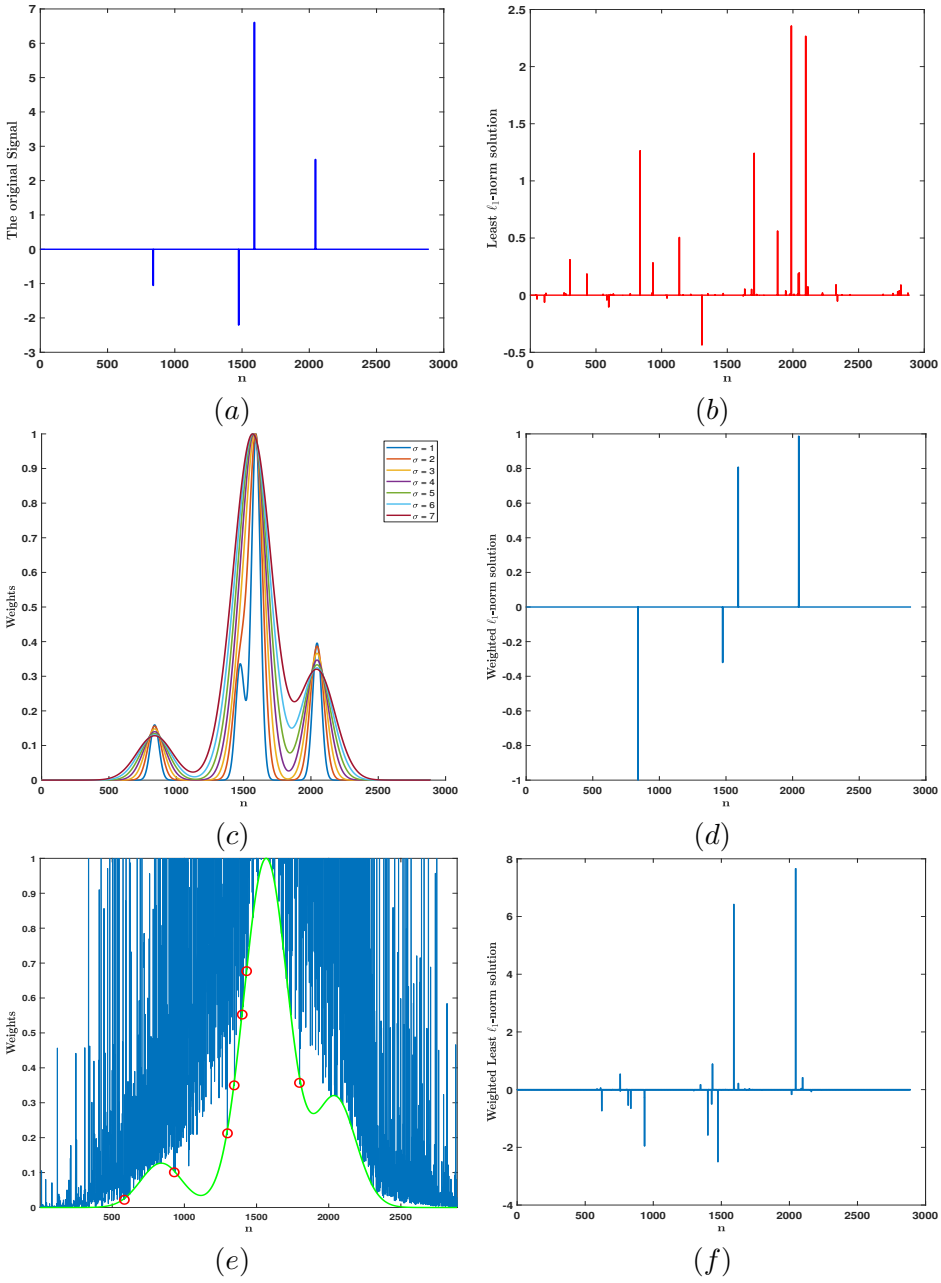


Figure 2.1: (a) Original source vector. (b) Least ℓ_1-norm solution. (c) Weights corresponding to $\sigma = 1, ..., 7$. (d) Least ℓ_1-norm solution of the weighted problem obtained with any of the weights shown in (c). (e) Weights for ($\sigma = 8$) (green). The blue line shows the upper bound on the weights, found by solving Eq. (2.16) (f) Weighted least ℓ_1-norm solution with the weights equal to the green curve in (e); this is an incorrect solution, arising because the weights exceed the upper bounds (see red circles in Fig. 1-e)

($\sigma = 8$). Figs. 2.1-a, 2.1-b, and 2.1-d show the true sparse signal, the least ℓ_1-norm estimate, and the weighted least ℓ_1-norm estimate using one of the weights shown in Fig. 2.1-c, respectively. Fig. 2.1-e shows the upper bound on the weights, obtained by solving (2.16) iteratively, while Fig. 2.1-f shows the weighted least ℓ_1-norm estimate for the weights of Fig. 2.1-e (i.e. $\sigma = 8$). One can see from Fig. 2.1-d that the weighted solution correctly estimates the support of the original vector for a large class of weights (Fig. 2.1-c), as long as the weights fall below the upper limits shown in Fig. 2.1-e (blue curves). Of course, at some point, when the support mismatch is large, i.e., $\sigma = 8$, the estimate deteriorates, as can be seen in Fig. 2.1-f.

2.3.3 Noisy Sparse Vector

To account for noise in the observations, we will minimize an objective function that is a tradeoff between $\|\mathbf{q}\|_1$ and the fitting error $\|\mathbf{y} - \mathbf{AWq}\|_2$, i.e.,

$$\underset{\mathbf{q}}{\text{minimize}} \quad h\|\mathbf{q}\|_1 + \|\mathbf{y} - \mathbf{AWq}\|_2, \tag{2.17}$$

where h is the regularization parameter. Let us first consider the case of the non-weighted problem corresponding to (2.17), i.e.,

$$\underset{\mathbf{x}}{\text{minimize}} \quad h\|\mathbf{x}\|_1 + \|\mathbf{y} - \mathbf{Ax}\|_2, \tag{2.18}$$

Assuming that Slater's and the strict complementary conditions hold, we give the following theorem.

Theorem 3. $\mathbf{x}^\star$ *is a solution to the problem of (2.18) if and only if there is* $\mathbf{u}$ *such that*

$$\begin{cases} (a)\ u_i = h & \text{if } x_i^\star > 0 \\ (b)\ u_i = -h & \text{if } x_i^\star < 0 \\ (c)\ |u_i| < h & \text{if } x_i^\star = 0 \end{cases}$$

(d) $\mathbf{u} = \mathbf{A}^T \frac{\mathbf{y}-\mathbf{Ax}^\star}{\|\mathbf{y}-\mathbf{Ax}^\star\|_2}$

Proof. See Appendix B. □

With Theorem 3, we can now state the effect of the weighting matrix $\mathbf{W}$ on the solution by the following theorem.

Theorem 4. *In the context of the problem of (2.17), let w_{ii} be the weight associated with q_i. If we choose $w_{ii} < \frac{h}{\|\mathbf{a_i}\|_2}$, then $q_i = 0$, where $\mathbf{a}_i$ is the column of $\mathbf{A}$ associated with q_i.*

Proof. See Appendix C. □

Theorem 4 suggests that, instead of solving the original problem, one can solve a reduced dimensionality problem by setting the entries in $\mathbf{q}$ that satisfy Theorem 4 condition to zero, and only solve for entries that do not satisfy the condition in Theorem 4. This will result in reducing the memory usage and calculation time when solving the problem in (2.17).

Here, we will give another interpretation of the condition in Theorem 4. Define the dual problem of primal problem in (2.18) as

$$\begin{aligned} \underset{\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2,\boldsymbol{\lambda}_3}{\text{maximize}} \quad & \mathbf{y}^T\boldsymbol{\lambda}_3 \\ \text{Subject to} \quad & \mathbf{A}^T\boldsymbol{\lambda}_3 + \boldsymbol{\lambda}_1 = \mathbf{1}h \qquad (19.1) \\ & \mathbf{A}^T\boldsymbol{\lambda}_3 - \boldsymbol{\lambda}_2 = -\mathbf{1}h \qquad (19.2) \\ & \boldsymbol{\lambda}_1 \succeq \mathbf{0}, \boldsymbol{\lambda}_2 \succeq \mathbf{0}, \|\boldsymbol{\lambda}_3\|_2 \leq 1. \end{aligned} \tag{2.19}$$

Let $\boldsymbol{\lambda}_3^\star$ be the solution to the dual variable associated with the objective function of the problem defined in (2.19). Theorem 4 states that for $\mathbf{x}_i^\star = 0$, $\boldsymbol{\lambda}_3^\star$ should lie between the two hyperplanes, defined as $\mathbf{a}_i^T\boldsymbol{\lambda}_3^\star = h$ and $\mathbf{a}_i^T\boldsymbol{\lambda}_3^\star = -h$, and for $|x_i^\star| > 0$, $\boldsymbol{\lambda}_3^\star$ should lie on one of the hyperplanes that has the same sign as $\mathbf{x}_i$.

Next, we study the effect of the weighting matrix $\mathbf{W}$ on the solution of (2.18). The constraints of the dual problem of (2.17) can be rewritten as

$$\begin{aligned} \|w_{ii}\mathbf{a}_i^T\boldsymbol{\lambda}_3^\star\|_\infty \leq h \\ \|\boldsymbol{\lambda}_3^\star\|_2 \leq 1. \end{aligned} \tag{2.20}$$

By assigning w_{ii} to small values, the distance between the two hyperplanes increases, and there might be no intersection of the two hyperplanes with the second norm of $\boldsymbol{\lambda}_3^\star$. When we assign high values for w_{ii}, the distance between the corresponding hyperplanes decreases, making these hyperplanes intersect with the second norm of $\boldsymbol{\lambda}_3^\star$.

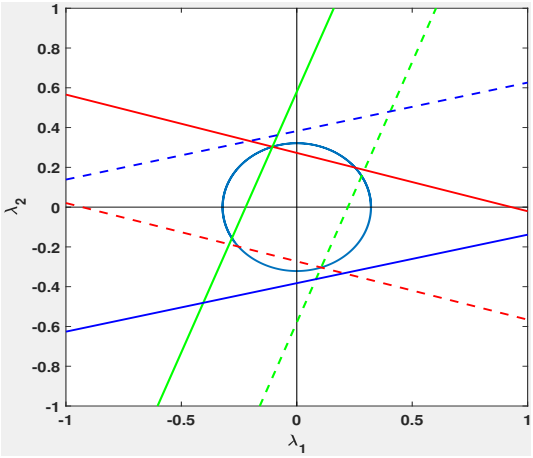


Figure 2.2: Solution space of $\boldsymbol{\lambda}_3 \in R^2$ in the non-weighted problem. Solid and dashed lines represent the hyperplanes associated with positive and negative h, respectively. Red, blue, and green lines are the hyperplanes associated with the first, second, and the third columns of $\mathbf{A}$, respectively.

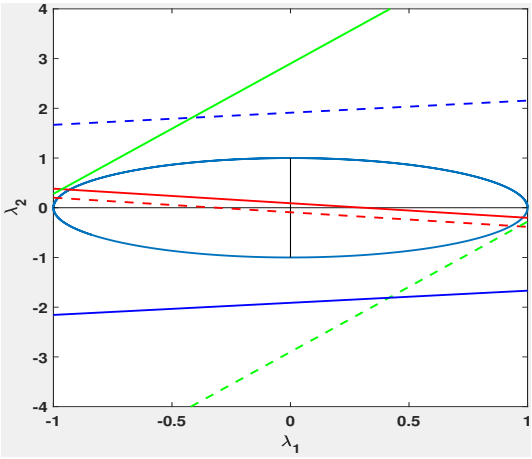


Figure 2.3: Solution space of $\boldsymbol{\lambda}_3 \in R^2$ for the weighted problem. Solid and dashed lines represent the hyperplanes associated with positive and negative h respectively. Red, blue, and green lines are the hyperplanes associated with the first, second, and the third columns of $\mathbf{A}$, respectively.

The above discussion is illustrated in Fig. 2.2 for a non-weighted case, and in Fig. 2.3 for the weighted case. In this example, we chose

$$A = \begin{bmatrix} 0.5377 & -2.2588 & 0.3188 \\ 1.8339 & 0.8622 & -1.3077 \end{bmatrix},$$

and $\mathbf{x}_0 = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T$. White noise was added to the output with signal to noise ratio (SNR) equal to 15 dB, and h was set to 0.5. In Fig. 2.2, the two hyperplanes associated with each column of $\mathbf{A}$ are plotted, with the solid line representing the hyperplane associated with positive h, dashed line the hyperplane associated with negative h. The red, blue, and green lines are the hyperplanes associated with the first, second, and third columns of $\mathbf{A}$, respectively. The blue circle represents the second norm of $\boldsymbol{\lambda}_3^\star$. As shown in Fig. 2.2, the value of $\boldsymbol{\lambda}_3^\star$ will be the point of intersection of the green and red lines with the $\boldsymbol{\lambda}_3^\star$ norm curve. The solution for $\boldsymbol{\lambda}_3^\star$ in this case indicates that $\mathbf{x}_3^\star$ will be zero, while $\mathbf{x}_1^\star$ and $\mathbf{x}_2^\star$ will not be zero. Indeed, the optimal solution is $\mathbf{x}^\star = \begin{bmatrix} 1.0209 & 0.0511 & 0 \end{bmatrix}^T$. Fig. 2.3 represents the solution of problem (2.17) with the same settings as in the above example, and for $\mathbf{W} = \text{diag}\begin{bmatrix} 3 & 0.2 & 0.2 \end{bmatrix}$. As shown in Fig. 2.3, the solution for $\boldsymbol{\lambda}_3^\star$ will be the intersection of the red hyperplanes with the unit norm circle. The solution in this case indicates that the second and third entries of $\mathbf{x}^\star$ will be zero, and only the first entry of $\mathbf{x}^\star$ will be non-zero. Thus, $\mathbf{x}^\star = \begin{bmatrix} 0.34 & 0 & 0 \end{bmatrix}^T$. The weighted approach is better in the sense that it has the real support of the actual sparse vector, while the non-weighted solution shows two active entries.

The distance between the two hyperplanes $w_{ii}\mathbf{a}_i^T\boldsymbol{\lambda}_3^\star = h$ and $w_{ii}\mathbf{a}_i^T\boldsymbol{\lambda}_3^\star = -h$ equals $\frac{2h}{w_{ii}\|\mathbf{a}_i\|_2}$. Based on the above discussion, by making the distance between these hyperplanes larger than two, and because $\|\boldsymbol{\lambda}_3^\star\|_2 \leq 1$, there will be no intersection between the unit norm ball and the above hyperplanes, which coincides with the result of Theorem 4.

The main result that can be concluded from the weighted noisy case is that by assigning high values to w_{ii}s corresponding to x_is that are expected to be non-zero, and small values to w_{ii}s corresponding to $\mathbf{x}_i$s that are expected to be zero, the solution to (2.17) will favor the solution $\mathbf{q}$ that has the same support as the underlying vector $\mathbf{x}_0$.

2.4 Application to EEG Sparse Source Localization

In this section, we apply the proposed approach to the problem of EEG source localization. EEG is a relatively inexpensive non-invasive neuroimaging technique, offering a window into the human brain function by measuring the electrical potentials over the scalp, which are reflective of the underlying neural activity. Compared to other non-invasive neuroimaging techniques, such as Functional Magnetic Resonance Imaging (fMRI), EEG offers superior temporal resolution, and hence continues to be an attractive imaging tool in several domains including basic neuroscience research [39–41], clinical neuroscience [42–44], and brain computer interfaces (BCIs) [45–48]. EEG however, suffers from the problem of poor spatial resolution due to volume conduction effect [49]. The signals recorded by EEG electrodes on the scalp surface represent a weighted sum of the electrical activity of the underlying neurons. As such, EEG recordings do not directly identify the location of sources in the brain.

Developing reliable EEG source localization techniques has been of great interest to the neuroscience and clinical communities, because of their potential in enabling an imaging tool with high accuracy in both the temporal and spatial domains. For example, EEG source localization could have important applications in noninvasive BCIs. The majority of existing EEG-based BCIs use information from the scalp-recorded signals (e.g. event-related desynchronization/synchronization (ERD/ERS)) for extracting features to distinguish actions [50]. Due to the poor spatial resolution of EEG, the degree of freedom of these BCIs has been limited to discriminating a small number (e.g. 6) of very "distinct" classes of actions [48], [51], [52], and as the number of classes increases, the classification performance degrades. This issue has been a major challenge for EEG-based BCIs, as in a realistic setting, more than few degrees of freedom are needed. Recent work [53], demonstrating improvement in the accuracy of discriminating several actions based on source domain as compared to sensor domain information, further motivates the application of source localization in EEG-based BCIs.

To date, several algorithms with different a priori constraints on sources have been proposed to solve the ill-posed inverse problem for EEG source localization [54, 55].

In MUltiple Signal Classification (MUSIC) [56, 57], Minimum Variance Beamforming (MVB) [58,59], and Linearly Constrained Minimum Variance (LCMV) methods [60,61], second order statistics are used to localize the sources. However, it is hard to obtain good estimates of those statistics, due to the non-stationarity of the data. For instance, in MVB, the number of statistically independent snapshots should be at least three times the number of channels to achieve stable source localization estimates [61], which would be a problem for systems with large number of channels.

Another approach of source localization methods is the class that fits the observations to a linear system model. The small number of obtained recordings at a given time, as compared to the internal mesh size of the brain [62] makes the source estimation problem underdetermined with infinitely many solutions [54]. Assuming that at a given time a small number of sources inside the brain are active above a certain threshold, the source localization problem can be formulated as a sparse signal recovery problem. EEG source localization methods that exploit the sparsity of $\mathbf{x}$ include ℓ_1-norm minimization [63–66], FOCUSS [20], MP [23], ORMP [23], and Source Deflated Matching Pursuit (SDMP) [67].

As already mentioned in Section slowromancapii@, strong equivalence conditions should be satisfied in order to use the ℓ_1-norm minimization for sparse EEG source localization. However, when using a realistic head model to construct the lead field matrix, these conditions may not be met, thus rendering the solution of ℓ_1-norm based methods possibly irrelevant. On the other hand, the problem of low SNR related to EEG source localization and the problem of local minimum can affect the performance of FOCUSS in correctly estimating the location of sources. Also, in all the greedy algorithms (i.e., MP, ORMP, and SDMP), the sparsity level should be known in advance, which is unrealistic in the EEG scenario. Further, the high correlation among the columns of the lead field matrix, as will be discussed in the following, represents another challenge for the greedy methods.

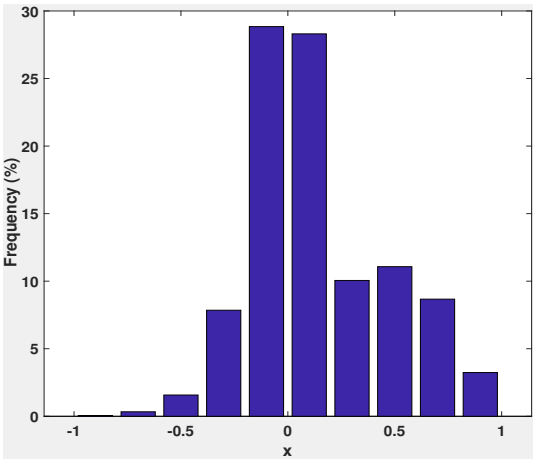


Figure 2.4: Histogram of normalized cross correlation of the columns of a realistic lead field matrix.

2.4.1 Model Description

The current-based dipole model [68], in which the active sources are modeled as dipoles was adopted here. By segmenting the cortex into m nodes, a dipole vector, called the lead field vector (LFV), is assigned to each node. With n number of electrodes on the scalp, at given time instant t, the EEG model can be described as

$$\mathbf{y}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{n_o}(t), \tag{2.21}$$

where $\mathbf{y}(t) \in R^n$ represents the electrode readings at time instant t, $\mathbf{x}(t) \in R^{3m}$ denotes the dipole source vector at time instant t, $\mathbf{A} \in R^{n\times 3m}$ is the lead field matrix and $\mathbf{n_o}(t) \in R^n$ denotes the noise vector. It is clear that the system described by (2.21) is underdetermined, i.e., for the same electrode readings $\mathbf{y}(t)$, infinite solutions for $\mathbf{x}(t)$ can be obtained.

The coherence histogram of the lead field matrix, obtained by the BrainStorm [38] toolbox, is shown in Fig. 2.4 (the diagonal elements are excluded). From the figure, one can see that approximately 15 % of the columns exhibit a correlation factor greater than 0.7, which indicates that there is no guarantee that the least ℓ_1-norm solution will coincide with the sparsest source vector. Also, from Fig. 2.4, one can see that there are vectors with correlation factor greater than 0.8, which violates the RIP condition [13], requiring that all k subsets columns should behave like orthonormal columns; k here represents the number of non-zero entries of the sparsest solution. Following the above

discussion, it is obvious that the lead field matrix does not satisfy the strong equivalence conditions described in Section slowromancapi@.

2.4.2 Simulation Setup

A 3-shell realistically-shaped head model (Colin27) [69], provided by the Brainstrom software package [38], was used to represent the geometry of the brain. The electrical conductivity for the cortex, skull, and scalp was set to 1.0 S/m, $\frac{1}{80}$ S/m, and 1.0 S/m, respectively.

The EEG is believed to be mainly generated by the inhibitory and excitatory post-synaptic potentials of cortical pyramidal nerve cells, which are spatially aligned perpendicular to the cortical surface [70]. Therefore, we only considered the cortex as the source space. While the activities from deep brain activations are generally believed to be poorly represented in EEG signals, if similar to [22], the model for relating measurements and sources can be approximated as linear and instantaneous, the proposed approach can also localize the sources if appropriate weights satisfying the conditions mentioned previously are used.

The cortex was divided into 966 grid points such that the mean distance between two grid points was 5 mm. 64 electrodes, following the International $10-10$ system of EEG sensor placement, were positioned in the sensor space. The lead field matrix was accordingly constructed [71]. This head model was used in both the construction of the forward model (calculating the electrode potentials in the sensor space from active dipoles in the source space), as well as in solving the inverse problem (reconstructing activity in the source space from electrode potentials in the sensor space). To simulate active sources, dipoles were modeled as sinusoidal signals with frequencies in the range of $[6-30]$ Hz. White noise was added to the observations obtained from the forward model. Our results and simulations are obtained on a single snapshot, so they do not depend on whether the sources are temporally independent of each other. However, obtaining the weighting matrix via the MUSIC method does exploit time correlations. Of course, one could use other methods to obtain the weighting matrix.

The inverse problem was solved using the proposed approach, the non-weighted

approach (i.e., (2.18)), FOCUSS, ORMP and MUSIC. To construct the weighting matrix $\mathbf{W}$ for the proposed approach, and also initialize FOCUSS, MUSIC was employed. If otherwise stated, 50 snapshots were used to estimate this matrix; this number of snapshots was selected because it resulted in comparable performance between MUSIC, FOCUUSS, and the proposed approach (see Fig. 2.9) for the case of 2 sources, 15 dB SNR, and 4 cm minimum distance between sources. Various conditions were considered (e.g. variable SNR, different number of sources). For each condition, 1000 Monte Carlo trials were performed, where in each trial, the locations of sources were chosen at random. For the case of multiple sources, the distance between active sources was kept to be greater than a predefined value, to ensure no overlap between estimated sources.

To evaluate the performance of different algorithms, the "success rate" was used as the performance metric. For k sources, we declared an estimation to be successful if the distance between the locations of the k largest estimated sources and the locations of the actual sources was less than a predefined threshold value (d). The success rate was defined as the ratio of the number of properly estimated source locations and the total number of sources. The selection of values for d was done along the lines of [67]. Considering that in our head model the mean distance between two grid points is 0.5 cm, setting d to 1 cm, will allow estimates located one grid point away from the true source to be considered as successful estimates. Note that if d is increased, the success rate from all algorithms is expected to increase, since a larger area around the true sources would qualify as correct estimation.

2.4.3 Simulation Results

In this section, we present simulation results to demonstrate the performance of the proposed method and compare it against existing sparse signal recovery methods [54]. Note that, given our grid size and $d = 1$ cm, for any two sources to be resolvable, the distance between them should be larger than 2 cm. In our simulations, we considered an additional 2 cm inter-source to ensure that there are no interference between sources.

Single Trial Simulation Results

The purpose of this simulation is to demonstrate the local minima problem and residual interference that are associated with FOCUSS and ORMP, respectively. In single trial simulations, two sources were considered at random locations with 4 cm minimum distance between them. The SNR was set to 25 dB. High SNR was selected here in order to reduce the effects of noise on the performance of FOCUSS estimation. Fig. 2.5 shows the results. As can be seen from Fig. 2.5-b, the estimation obtained by the non-weighted approach contains additional sources that do not correspond to real sources (Fig. 2.5-a). The estimate of ORMP (Fig. 2.5-d) suffers from residual interference [67]. Fig. 2.5-e and -f, show respectively the estimated vector by FOCUSS and the proposed approach, using the weights of Fig. 2.5-c. Although the weights assign high values to the support of the underlying sparse vector, the estimate via FOCUSS shows false active sources, due to the local minima problem. Fig. 2.5-f shows the estimate obtained via the proposed approach; one can see that, as compared to ORMP and FOCUSS, a more accurate estimation of sources is obtained.

Monte-Carlo Simulation Results

Fig. 2.6 shows the performance of the proposed approach, FOCUSS, ORMP, non-weighted, and MUSIC versus SNR for the case of two sources, where the minimum distance between the sources equals 4 cm. One can see that the proposed approach is more robust than the other methods. At SNR of 0 dB, the success rate for the proposed method is 80%, while for FOCUSS is 30%. The low success rate for FOCUSS at low SNR is because of its iterative procedure which involves finding the inverse of $\mathbf{AW}$. This process is sensitive to noise due to the ill-posed lead field matrix $\mathbf{A}$ [22].

Figs. 2.7 and 2.8 compare the performance of all five algorithms for different number of sources at SNR of 10 dB, with the minimum distance between the sources set to 4 cm, and 8 cm, respectively. While the proposed approach offers superior performance as compared to others when the number of sources is smaller than 5, as the number of sources increases, the performance of all algorithms degrades. Comparing Figs. 2.7 and 2.8, it can be seen that as the distance between sources increases, the performance

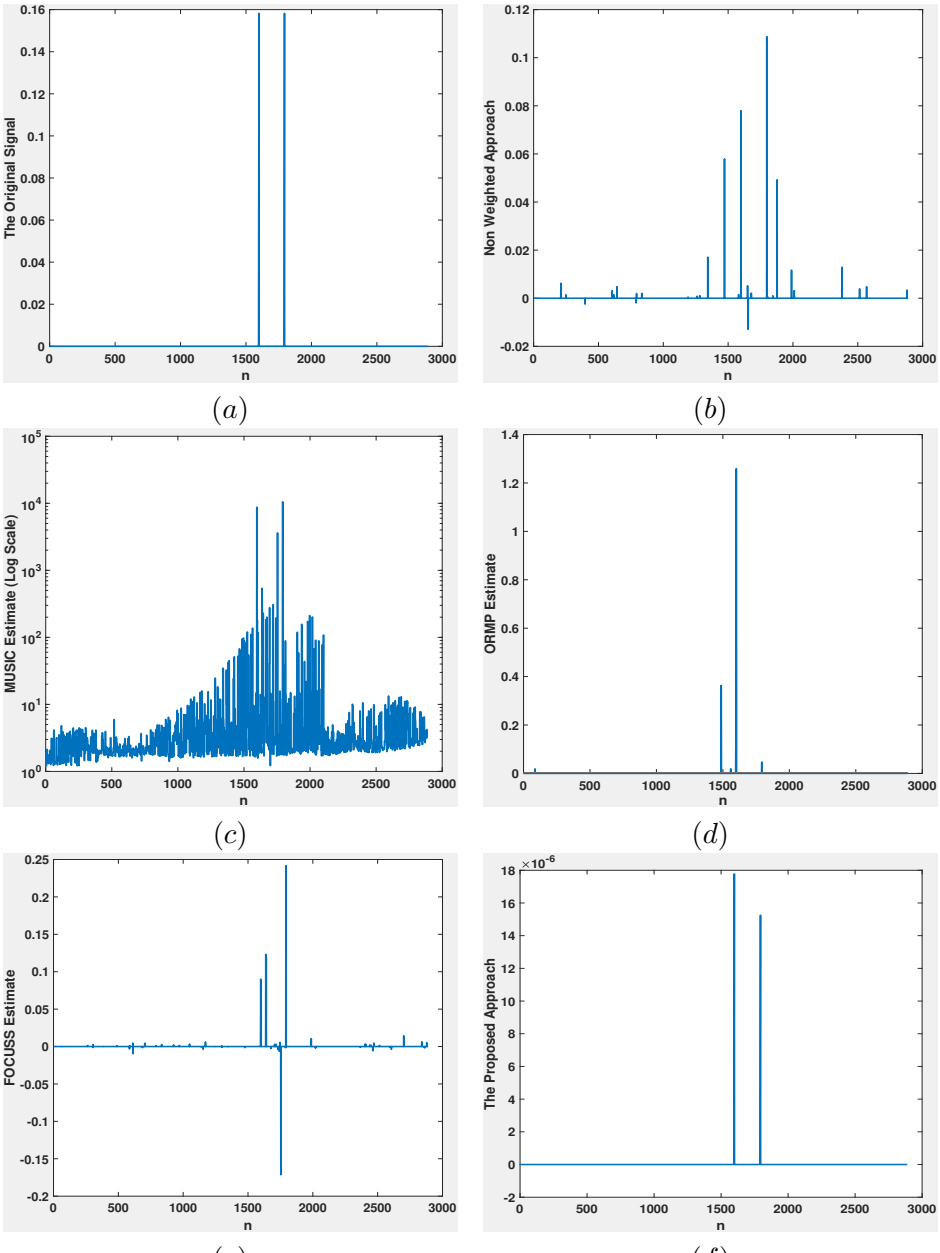


Figure 2.5: (a) The original source vector (2 sources, 25 dB SNR, and 4 cm minimum distance). (b) Estimate obtained from the non-weighted approach. (c) Estimate obtained from MUSIC shown in log scale. (d) Estimate obtained from ORMP. (e) Estimate obtained from FOCUSS. (f) Estimate obtained from the proposed approach.

improves. This could be due to the fact that distant sources correspond to the columns of the lead field matrix that are less correlated.

Fig. 2.9 shows the success rate of the proposed approach, FOCUSS, and MUSIC for the case of two sources and SNR of 15 dB, when different number of snapshots are used to construct the weighting matrix. One can see that as the number of snapshots decreases, the success rate of all approaches degrades, with the proposed approach performing better than FOCUSS. The performance degradation is due to the degradation in MUSIC estimate used to construct the weighting matrix.

Table 2.1: Performance Comparison of the proposed approach with other Methods. For each scenario, 1000 Monte Carlo trials for 2 sources at SNR=10 dB are considered.

Method	**Average Processing Time** (sec)	**Success Rate** (%) **for Low Coherence A**	**Success Rate** (%) **for Signal in the range of** (0.5-50)Hz
MUSIC	0.0725	100	88.78
FOCUSS	1.4296	100	86.84
Proposed Approach	1.991	100	97.24
ORMP	0.1229	100	75.5
Non-weighted	1.9270	93.27	31.9

In the case of a sensing matrix with low coherence, the performance of the proposed approach will be similar to the other methods. To demonstrate this, we performed a simulation for two sources, SNR of 10 dB, and a Gaussian matrix (the mutual coherence was 0.578) as the dictionary matrix. The results are summarized in Table I, third column. the above simulations we considered signals as dipoles with frequencies in the range of $[6-30]$ Hz. To see the effect of the frequency range, here, we also consider a scenario where the frequency range for signal generation is $[0.5-50]$ Hz. The results, summarized in Table I, last column, are similar to those of Fig. 2.6 for SNR of 10 dB, suggesting that the choice of frequency range for generating signals does not affect the performance of the localization algorithms.

To compare the computational complexity of different approaches in terms of average processing time per trial, 1000 Monte Carlo trials for the case of 2 sources, and SNR of 10 dB were conducted. The results are shown in Table I. The ℓ_1-norm-based approaches (both weighted and non-weighted) as compared to FOCUSS, MUSIC, and ORMP, are more computationally intensive, as expected.

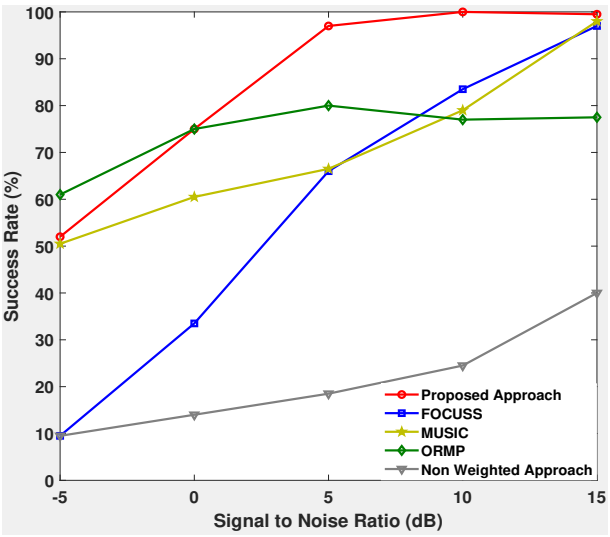


Figure 2.6: Success rate of the proposed approach, FOCUSS, ORMP, non-weighted, and MUSIC for the case of two sources as a function of SNR. The minimum distance between sources was set to be 4 cm.

2.4.4 Experimental Results

We also examined the performance of the proposed approach in solving the source localization problem, using real EEG data. EEG experiment for eliciting auditory evoked potentials (AEPs) [67] was conducted with one volunteer, who provided his written informed consent. The stimulus was a pure tone of 1000 Hz with duration of 40 ms, that was presented to the left ear of the participant. The paradigm consisted of 1092 trials with inter stimulus interval (ISI) of 760 ms. Brain activities were recorded using a 64-channel EEG system (Brain Products, Germany) with 1 kHz sampling rate.

Preprocessing was performed using EEGLAB [72]. Recorded signals were first down-sampled to 256 Hz to reduce the processing time when performing independent component analysis (ICA) for the artifact removal step. EEG recordings were bandpass filtered between 0.5 Hz and 100 Hz with a notch filter at 60 Hz. Bad channels were identified (five channels: FP1, FP2, AF7, FT9, and FT7), and their corresponding signals were replaced with the average of the signals from their neighboring channels. ICA was then employed to remove the artifacts (e.g. eyeblinks). The covariance matrix was estimated using the samples that correspond to 50 ms before the stimuli, and was used for whitening the data. Epochs were extracted and then averaged across trials.

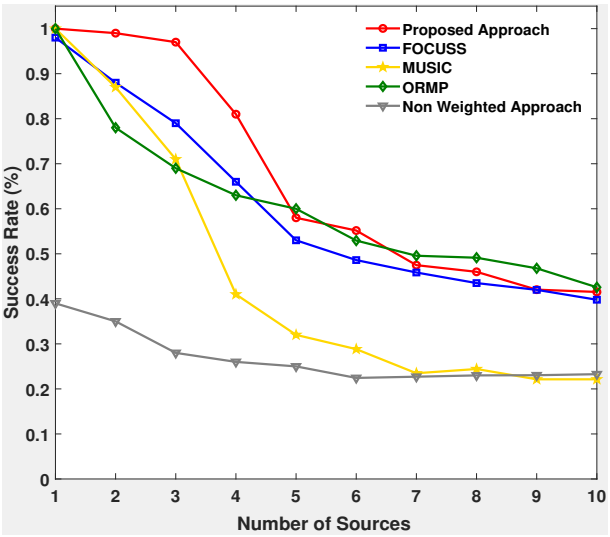


Figure 2.7: Success rate of the proposed approach, FOCUSS, ORMP, non-weighted, and MUSIC as a function of number of sources. SNR was set at 10 dB, and the minimum distance between sources was set to be 4 cm.

It has been reported that the most relevant components associated with the auditory experiments are P50 and N100 [73]. The Event related potential (ERP) waveforms for all channels are shown in Fig. 2.10 (filtered down to 30 Hz for display) where both P50 and N100 components can be identified. This result is aligned with what has been reported in other EEG studies [74–76]. While here the results for N100 are presented, the localization can also be performed for P50 or other ERPs of interest.

To estimate the location of activities related to N100, the segment $[100 - 132]$ ms from ERP was selected. Since the exact location of sources are unknown, to compare the performance of different localization methods, we take a qualitative approach [77], with reference to the existing knowledge about the expected active regions corresponding to N100. As reported in [67, 78], for this task and at N100, activations in both left and right primary auditory cortices are expected to occur.

To construct the weighting matrix $\mathbf{W}$, for the proposed approach and for initializing FOCUSS, MUSIC with 8 snapshots (about 32 ms) was used. The number of sources (sparsity level) for ORMP was restricted to 10. For each localization method, the inverse problem was solved at each sample over the duration of $[100 - 132]$ ms, and the average of the estimated sources across all time samples was obtained. Top, right and left views of the cortex, indicating the estimated locations of active sources based

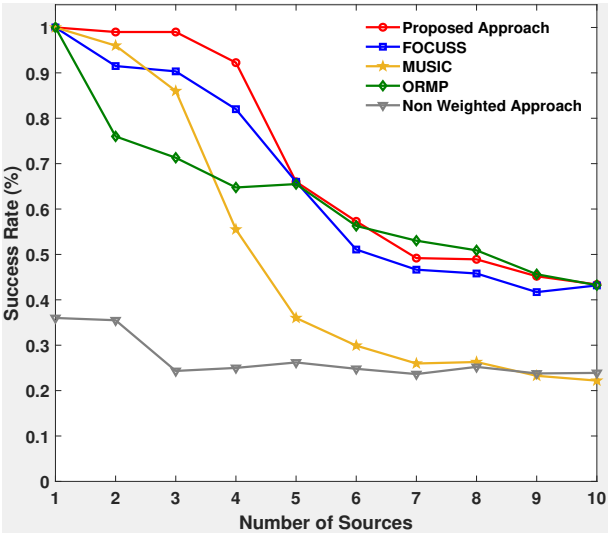


Figure 2.8: Success rate of the proposed approach, FOCUSS, ORMP, non-weighted, and MUSIC as a function of number of sources. SNR was set at 10 dB, and the minimum distance between sources was set to be 8 cm.

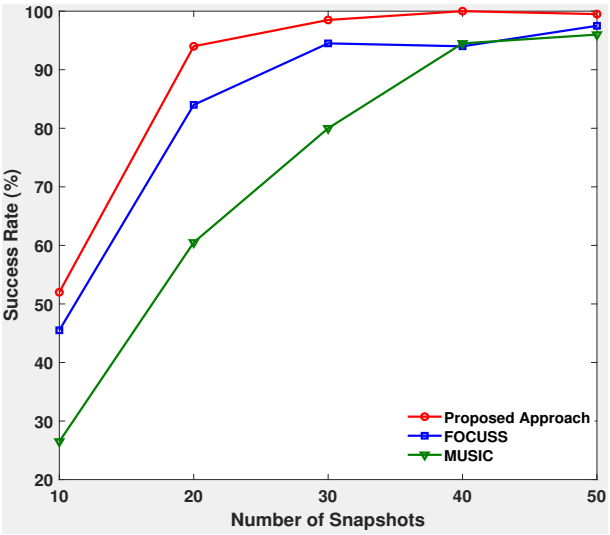


Figure 2.9: Success rate of the proposed approach, FOCUSS and MUSIC for the case of two sources and SNR of 15 dB, for different number of snapshots used to construct the weighting matrix.

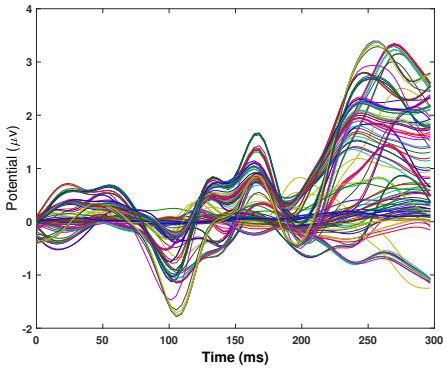
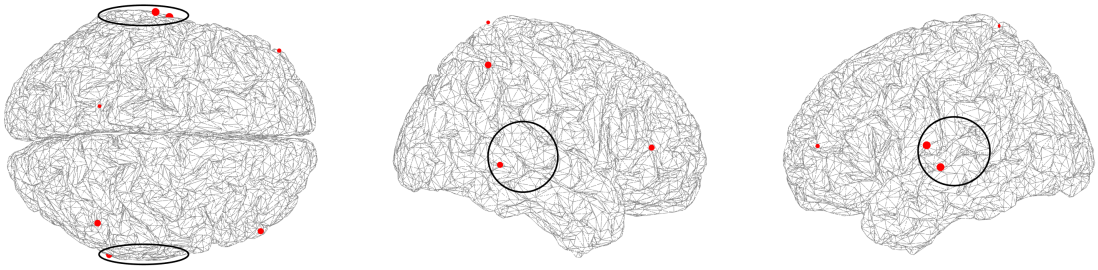


Figure 2.10: ERP in response to an auditory task for all 64 channels.

on the proposed approach, FOCUSS, and ORMP are shown in Figs. 2.11, 2.12, and 2.13, respectively. The proximity of the locations of the primary auditory cortices is also shown in Figs. 2.11-2.13 via black circles. The proposed approach identifies active sources in both left and right auditory cortices. This result is aligned with previous fMRI studies [79]. Few active sources are also identified in other regions of the brain. Brain activations related to N100, located outside the auditory cortex, have also been reported in previous studies [67], [80]. FOCUSS shows activations near the temporal lobes, as well as several other regions in the brain. Deviation from the expected active regions in FOCUSS could be due to the local minima problem that associated with FOCUSS or the low SNR [62]. Previous studies have shown variations in features of auditory-based ERPs (e.g. amplitude of N100) for different individuals [74]. The observed low SNR could therefore, be due to the fact that the obtained ERP is from one subject. Due to high coherence possessed by the realistic lead field matrix, the performance of ORMP is expected to be degraded. This is also observed in Fig. 2.13, where ORMP fails to localize active sources in expected regions.

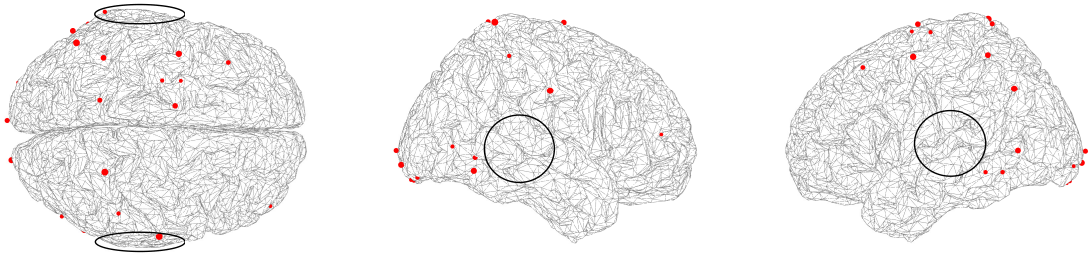
2.5 Conclusions

A weighting approach for sparse signal support estimation has been presented. We have shown that by appropriately selecting the weights, we can formulate an ℓ_1-norm minimization problem that satisfies the RSP, even if the original problem does not satisfy the RSP. Conditions on the weights for both noise free and noisy cases have



(a) (b) (c)

Figure 2.11: Estimated location of sources in the cortex via the proposed approach, a) top view, b) right view, and c) left view.



(a) (b) (c)

Figure 2.12: Estimated location of sources in the cortex via FOCUSS, a) top view, b) right view, and c) left view.

(a) (b) (c)

Figure 2.13: Estimated location of sources in the cortex via ORMP, a) top view, b) right view, and c) left view.

been provided. Although those conditions involve information about the support of the sparse vector, the class of good weights is very wide, and in most cases encompasses estimates obtained via a conventional method that does not encourage sparsity. Simulation results have shown that in practical scenarios, using the proposed approach with weights constructed based on estimate obtained via a conventional method that does not encourage sparsity, results in significantly improved localization of the sparse signal samples as compared to directly applying ℓ_1-norm minimization. As an application example, we applied the proposed approach to the EEG source localization problem. Simulated and real EEG data have been considered and the proposed approach was applied using MUSIC as an estimate to construct the weights. The impact of SNR and number of active sources on the performance of the proposed approach have been studied through Monte Carlo simulations, and the results have been compared with those of FOCUSS, ORMP and the non-weighted approach. The proposed approach appears to be more robust in terms of SNR as compared to FOCUSS, and in terms of SNR and the number of sources as compared to ORMP and non-weighted approach. Using EEG data, we have also qualitatively evaluated the performance of the proposed approach in localizing active sources within primary auditory cortices that are responsible for auditory N100. While for the purpose of validation here we had experimental data from one subject, our future work will involve the inclusion of more subjects.

Chapter 3

Weighted Sparse Bayesian Learning (WSBL) for Basis Selection in Linear Underdetermined Systems

3.1 Introduction

Sparse vector recovery problems arise in many applications including biomedical imaging [1–5], and radars [6–11]. The goal there is to find the sparse vector with the smallest number of non-zero entries that give rise to the observed vector. Mathematically, the problem is formulated as an ℓ_0-norm minimization problem. However, due to the associated complexity [12], the relaxed ℓ_1-norm minimization problem is usually solved instead. The solution of the ℓ_0-norm minimization problem coincides with that of ℓ_1-norm minimization problem if certain conditions are satisfied, including the Restricted Isometry Property (RIP) [13], the Null Space Property (NSP) [14], the Mutual Coherence [15], or the Range Space Property (RSP) [16]. In practice, however, the aforementioned conditions may not be met, in which case the least ℓ_1-norm solution is not related to the sparsest solution. Re-weighted iterative approaches have also been proposed for sparse vector estimation. In FOCal Underdetermined System Solver (FOCUSS) [17] a reweighted ℓ_2-norm minimization problem is solved, making use of initial weights. Although FOCUSS has been shown to converge, it may converge to a local minimum [17]. In [18], a reweighted iterative ℓ_1-norm algorithm for enhancing sparsity is proposed. However, no convergence guarantees are provided in [18]. For cases in which the ℓ_1-norm minimization problem does not yield the least ℓ_0-norm solution, a weighted approach was proposed in [2], which exploits available information, such as a low resolution estimate of the sparse vector, to bias the solution and obtain the

The work was supported by NSF under grant ECCS 1408437.

underlying sparse vector.

Another class of sparse vector recovery methods are the probabilistic approaches, in which the estimated sparse solution is obtained by maximizing the Bayesian posterior using sparsity inducing priors [26–28]. In Sparse Bayesian Learning (SBL), Gaussian priors with distinct variances, called hyperparameters, are assigned to the entries of the sparse vector. These hyperparameters are assumed to follow Gamma distribution with shape parameter a and rate parameter b. These variances are estimated by maximizing the marginal likelihood function. Unlike [18], it was shown that SBL always converges [19], thanks to the well known properties of Expectation-Maximization. SBL has been proposed for classification [26], regression [26], and sparse basis selection [19]. In [29], support knowledge-aided sparse Bayesian approach was proposed for applications where erroneous prior knowledge of the support is available. Depending on the erroneous support, different update rules are assigned to the corresponding parameters b_is. The b_is that belong to the erroneous support are considered as random variables, and their corresponding parameters b_is are estimated along with the hyperparameters. Low values are assigned to b_is that do not belong to the erroneous support. All the above aforementioned methods rely on a single snapshot for sparse vector recovery. In [30], Multiple response Sparse Bayesian Learning (MSBL) was proposed as an extension to SBL for multiple snapshots case, in which a sparse vector is recovered assuming stationary sources across the snapshots. When noise is present, SBL estimates the noise variance as well as the hyperparameters. On the other hand, non-probabilistic methods, such as Basis Pursuit De-Noising (BPDN) [31, 32], reweighted ℓ_1-norm [18], and weighted ℓ_1-norm [2], assume the noise variance to be known in advance.

Here, we propose Weighted SBL (WSBL) for sparse signal recovery, which relies on a single snapshot. Unlike SBL, where all hyperparameter priors follow the Gamma distributions with the same parameters, in WSBL, the hyper-parameters have a distinct set of parameters. This introduces more degrees of freedom of this optimization and thus improves accuracy. These parameters act as weights, giving more importance to some hyperparameters over others. The relative importance of the hyperparameters can be determined based on a low-resolution estimate of the sparse vector, for example

an estimate obtained via a method that does not encourage sparsity. The choice of the MUltiple SIgnal Classification (MUSIC) estimate as weight are analyzed. An important advantage of WSBL is that it makes it easy to select the threshold that distinguishes between zero and non-zero elements in the recovered sparse vector, which reduces the convergence time of WSBL.

Theoretical analysis based on variational approximation theory, and also simulation results demonstrate that WSBL results in substantial improvement in terms of probability of detection and probability of false alarm, as compared to SBL, especially in the low signal to noise ratio regime. Our work is different than [29] in the following points. First, in our approach, the values of b_is are constant through out the entire estimation process, which is in contrast to the work in [29], where the values of b_is that belong to the support are re-estimated in each iteration. Also, in our approach, both parameters a_is and b_is are assigned different values depending on the weight vector, while in [29], only b_is that belong to the support are assigned different values for different entries.

The proposed approach is a good candidate for target estimation in Multiple Input Multiple Output (MIMO) radar, a technology that has received noticeable attention in recent years, for achieving high resolution with fewer antennas as compared to conventional radars [81, 82]. MIMO radar with millimeter wave technology is currently being evaluated for applications in Advanced Driver Assistance Systems (ADAS), like Automatic Emergency Breaking (AEB) [83]. With mm-wave technology, the range cell is small, thus, it is reasonable to expect that there is only a small number of targets in the range cell. In other words, the target scene is sparse, and target estimation can be formulated as a sparse signal recovery problem. The proposed WSBL is applied to estimate DOA of targets for the colocated MIMO radar scenario, and its performance is compared to well known Bayesian approaches.

Preliminary results of this work appeared in [84]. Here, in addition to the results of [84], we provide theoretical analysis to justify the performance improvement of WSBL as compared to SBL, especially in the low SNR regime. Also, we provide an upper limit on the hyperparameters, which makes it easier to select a threshold for distinguishing between zero and non-zero elements in the recovered sparse vector. Also, we validate

the use of MUSIC [56] estimate as a weight vector for the proposed approach. Finally, we present results for applying the proposed approach for a Multiple Input Multiple Output (MIMO) radar scenario. The contributions of our work can be summarized as improving the performance of sparse vector recovery, achieving faster convergence time, and robustness in noise as verified by both theoretical analysis and simulations.

The chapter is organized as follows. In Section 3.2 we discuss some background on SBL, as proposed in [26] and [19]. Section 3.3 proposes the WSBL approach, discusses an important property on the weight vector associated with the values of hyperparameters after convergence, and discusses use of MUSIC estimate as a weight vector in WSBL. Section 3.4 extends the results for the complex case scenario. Section 3.5 discusses the SBL performance at low signal to noise ratio (SNR), and Section 3.6 uses variational approximation theory to explain why WSBL outperforms SBL. Section 3.7 presents simulation results, while Section 3.8 provides concluding remarks.

3.2 Background Theory

Consider the linear underdetermined system

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n}, \tag{3.1}$$

with the elements of $\mathbf{n} \in \mathbf{R}^m$ being independent identically distributed (i.i.d.), zero-mean, Gaussian, with variance σ^2; $\mathbf{A}$ an $m \times n$ dictionary matrix with $n >> m$; $\mathbf{x} \in \mathbf{R}^n$ the sparse vector to be estimated; and $\mathbf{y} \in \mathbf{R}^m$ the vector of observations. The probability density function (pdf) of $\mathbf{y}$ given $\mathbf{x}$ and σ^2 is

$$p(\mathbf{y} \mid \mathbf{x}, \sigma^2) = (2\pi\sigma^2)^{-m/2} \exp(-\frac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2). \tag{3.2}$$

Assuming that the entries of $\mathbf{x}$ are i.i.d. Gaussian $\mathcal{N}(0, \alpha_i^{-1})$, the conditional pdf of $\mathbf{x}$ equals

$$p(\mathbf{x} \mid \boldsymbol{\alpha}) = \prod_{i=1}^{n} \left(\frac{\alpha_i}{2\pi}\right)^{1/2} \exp\left(\frac{-x_i^2\alpha_i}{2}\right), \tag{3.3}$$

where $\boldsymbol{\alpha}$ is a column vector, whose entries, α_is, are referred to as the hyperparameters [19,26]. The values of α_is are the reciprocal of the corresponding variances of x_is. When

α_i is high, the distribution of the corresponding x_i is highly concentrated around zero, giving high probability for x_i to be zero.

On taking the α_is to be i.i.d Gamma-distributed with parameters a and b [26], the pdf of $\boldsymbol{\alpha}$ is

$$p(\boldsymbol{\alpha}) = \prod_{i=1}^{n} \text{Gamma}(\alpha_i; a, b), \tag{3.4}$$

with $\text{Gamma}(\alpha; a, b) = \Gamma(a)^{-1} b^a \alpha^{a-1} e^{-b\alpha}$, and $\Gamma(a) = \int_0^\infty t^{a-1} e^{-t} dt$. The Gamma distribution parameters a and b are referred to as shape and rate parameters, respectively. For small a, the Gamma pdf is concentrated around zero, while for large a, the Gamma pdf is concentrated around its mode, i.e., $\frac{a-1}{b}$. The parameter b influences the shape of the pdf; a larger b has the effect of compressing the pdf, while a smaller b expands the pdf. On also taking σ^{-2} to be Gamma distributed with parameters c and d, [26] i.e.,

$$p(\sigma^{-2}) = \text{Gamma}(\sigma^{-2}; c, d), \tag{3.5}$$

the posteriori pdf $p(\mathbf{x}, \boldsymbol{\alpha}, \sigma^2 \mid \mathbf{y})$ can be written as

$$p(\mathbf{x}, \boldsymbol{\alpha}, \sigma^2 \mid \mathbf{y}) = p(\mathbf{x} \mid \mathbf{y}, \boldsymbol{\alpha}, \sigma^2) p(\boldsymbol{\alpha}, \sigma^2 \mid \mathbf{y}). \tag{3.6}$$

According to [26], the $p(\mathbf{x}|\mathbf{y}, \boldsymbol{\alpha}, \sigma^2)$ is Gaussian distributed with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ equal to

$$\boldsymbol{\mu} = \sigma^{-2} \boldsymbol{\Sigma} \mathbf{A}^T \mathbf{y}, \tag{3.7}$$

$$\boldsymbol{\Sigma} = (\sigma^{-2} \mathbf{A}^T \mathbf{A} + \mathbf{F})^{-1}, \tag{3.8}$$

where $\mathbf{F} = \text{diag}(\alpha_1, \alpha_2, ..., \alpha_n)$. Following the same approximation as in [26], $p(\boldsymbol{\alpha}, \sigma^2 | \mathbf{y})$ is approximated by a delta function at its mode. This approximation decomposes the problem of maximizing (3.6) into two distinct problems: maximizing $p(\boldsymbol{\alpha}, \sigma^2 \mid \mathbf{y})$ with respect to $\boldsymbol{\alpha}$ and σ^2, and maximizing $p(\mathbf{x} | \mathbf{y}, \boldsymbol{\alpha}, \sigma^2)$ with respect to $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ using (3.7) and (3.8).

Regarding the first problem, it holds that [26]

$$p(\boldsymbol{\alpha}, \sigma^2 \mid \mathbf{y}) \propto p(\mathbf{y} \mid \boldsymbol{\alpha}, \sigma^2) p(\boldsymbol{\alpha}) p(\sigma^2), \tag{3.9}$$

where $\boldsymbol{\alpha}$ and σ^2 are assumed to be independent, and $p(\mathbf{y}\,|\,\boldsymbol{\alpha}, \sigma^2)$ is Gaussian with mean zero, and covariance matrix $(\sigma^2\mathbf{I} + \mathbf{A}\mathbf{F}^{-1}\mathbf{A}^T)^{-1}$. An iterative approach can be used to estimate the values of $\boldsymbol{\alpha}$ and σ^2 that maximize (3.9). For $\boldsymbol{\alpha}$, differentiating (3.9) with respect to $\log(\alpha_i)$, and equating to zero, the new value of α_i can be written as

$$\alpha_i^{(\text{new})} = \frac{1 + 2a}{\mu_i^2 + \Sigma_{ii} + 2b}. \tag{3.10}$$

For σ^2, differentiating with respect to $\log(\sigma^{-2})$ and equating to zero, the update rule for σ^2 can be written as

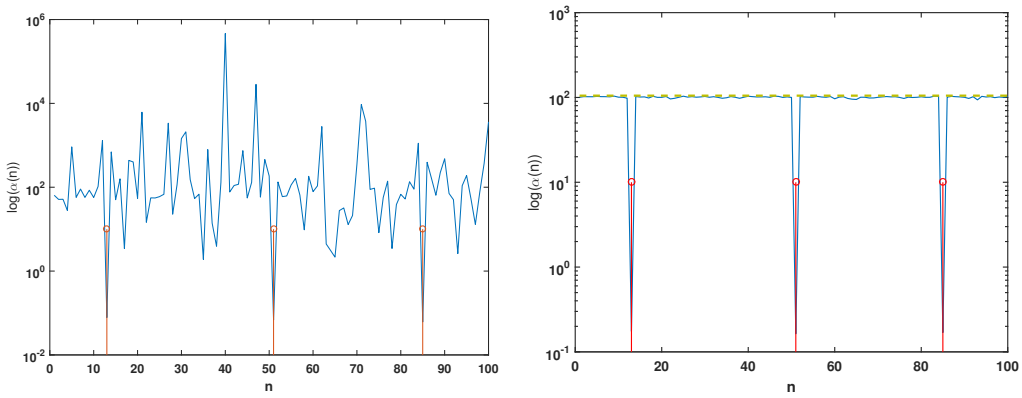
$$(\sigma^2)^{(\text{new})} = \frac{\|\mathbf{y} - \mathbf{A}\boldsymbol{\mu}\|_2^2 + 2d}{N - \sum_{i=1}^{n} \gamma_i + 2c}. \tag{3.11}$$

with $\gamma_i = 1 - \alpha_i \Sigma_{ii}$. In summary, the maximization of (3.6) is implemented by updating the posterior covariance and the mean via (3.7) and (3.8), and then updating the values of $\boldsymbol{\alpha}$ and σ by applying (3.10) and (3.11). The above procedure continues until a convergence criterion is satisfied [26].

To better explain the sparse nature of the estimated vector via SBL, let us find the pdf of x_i by averaging out hyperparameter α_i, i.e., $p(x_i) = \int p(x_i \mid \alpha_i) p(\alpha_i) d\alpha_i$. If $a = b \approx 0$, the result of the previous integration is an improper distribution, that is inversely proportional to $|x_i|$ [19, 26]. Such distribution induces sparsity, as it is highly peaked at zero.

In the Bayesian Sparse recovery approach, the α_is influence the basis selection in the estimation of the sparse vector $\mathbf{x}$. In SBL, it was shown in [26] that after convergence, most of the α_is tend to have very large values. This implies that the variance of $p(x_i \mid \mathbf{y}, \boldsymbol{\alpha}, \sigma^2)$ is highly peaked at zero, which corresponds to $x_i = 0$ with high probability.

SBL, by default assumes that all the entries of $\mathbf{x}$ have equal probabilities to belong to the support of the underlying estimated sparse vector. However, in some applications, some rough estimate of the underlying sparse vector is available. In the next section, we show how one can use such available estimate to differentially treat the entries of $\mathbf{x}$, which is the basis of our proposed approach.



(a) (b)

Figure 3.1: Hyperparameters after convergence in (a) SBL and (b) WSBL. The red lines represent the actual support oFf the original sparse vector. The yellow line represent the theoretical upper limit of the hyperparameters in WSBL.

3.3 The Proposed Weighted SBL (WSBL) Approach

For the proposed approach, we still consider the α_is to be independent, Gamma distributed, but with parameters a_i and b_i. Thus, the pdf of $\boldsymbol{\alpha}$ equals

$$p(\boldsymbol{\alpha}) = \prod_{i=1}^{n} \text{Gamma}(\alpha_i \mid a_i; b_i). \tag{3.12}$$

The posterior pdf $p(\mathbf{x}, \boldsymbol{\alpha}, \sigma^2 | \mathbf{y})$ can also be described as in (3.6), where $p(\mathbf{x}|\mathbf{y}, \boldsymbol{\alpha}, \sigma^2)$ is a Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ as in (3.7) and (3.8), respectively. Along the lines of SBL [26], we approximate $p(\boldsymbol{\alpha}, \sigma^2 | \mathbf{y})$ as a delta function at its mode. Then, the problem of maximizing $p(\mathbf{x}, \boldsymbol{\alpha}, \sigma^2 \mid \mathbf{y})$ can be redefined as that of maximizing (3.9), with $p(\boldsymbol{\alpha})$ as given in (3.12). On differentiating (3.9) with respect to $\log(\alpha_i)$, and equating to zero we get the following update rule

$$\alpha_i^{(\text{new})} = \frac{1 + 2a_i}{\mu_i^2 + \Sigma_{ii} + 2b_i}. \tag{3.13}$$

Suppose that we have a coarse estimate $\mathbf{w}$, that has large values along the support of $\mathbf{x}$, and low but nonzero values elsewhere. On assigning $a_i = \frac{1}{w_i}$, and $b_i = w_i$, if w_i is large, the corresponding distribution of α_i is highly peaked at zero. Thus, the corresponding x_i is nonzero with high probability. On the other hand, if w_i is small, the distribution of the corresponding α_i peaks around its mode, and thus the corresponding x_i is zero with high probability. We refer to the proposed approach as Weighted SBL

(WSBL). The estimation of $\boldsymbol{\alpha}, \sigma^2$, and $\mathbf{x}$ is achieved in an iterative manner. Initially, the hyperparameters are assigned to values less than the threshold η. In each iteration, the values of $\boldsymbol{\alpha}, \sigma^2$ are updated using (3.13) and (3.11), respectively. Using the updated $\boldsymbol{\alpha}$ and σ^2, $\boldsymbol{\Sigma}$ and $\boldsymbol{\mu}$ are updated by applying (3.8) and (3.7), respectively. The hyperparameters with values greater than η are excluded from the next iteration, and their corresponding x_is are set to zero, i.e., α_i is compared to a threshold η, and if

$$\frac{1 + 2/w_i}{\mu_i^2 + \Sigma_{ii} + 2w_i} > \eta, \tag{3.14}$$

then the corresponding x_i is set to zero in future iterations.

3.3.1 About the Weight Vector

Here, we discuss an important property of the weight vector $\mathbf{w}$, associated with the proposed WSBL approach, which is related to the final values of α_is after convergence. Given the weights vector $\mathbf{w}$, the α_is are bounded from above, as described in the following proposition.

Proposition 1. *Let w_{is} be the smallest entry in $\mathbf{w}$. After convergence of WSBL, the α_is are bounded as follows:*

$$0 \leq \alpha_i \leq \frac{w_{is} + 2}{2w_{is}^2} \tag{3.15}$$

Proof: Since μ_i^2 and Σ_{ii} are always positive, one can see from (3.13) that the largest value of α_i is smaller than $\frac{1+2a_i}{2b_i}$. By construction, we have $a_i = \frac{1}{w_i}$ and $b_i = w_i$. Substituting a_i and b_i in $\frac{1+2a_i}{2b_i}$, we get $\frac{w_i+2}{2w_i^2}$. The maximum value of α_i is attained when $w_i = w_{is}$, which proves the proposition. ■

The upper bound of α_i makes it easier to set a threshold η for differentiating between zero and non-zero entries in $\mathbf{x}$, i.e., we can select $\eta \in (0, \frac{w_{is}+2}{2w_{is}^2})$ and decide that $x_i = 0$ if $\alpha_i > \eta$, or $x_i \neq 0$ if $\alpha_i < \eta$. On the other hand, in SBL, the values of αs are not bounded, making it difficult to choose such a threshold. The threshold plays a big role in the final estimate, since all α_is greater than the threshold are excluded from the final estimation, and considered as zeros in $\mathbf{x}$ [26]. We should emphasize that in Proposition 1, we assumed that the minimum of $\mathbf{w}$ is greater than zero. In other words, we should

use a weight vector that is not sparse. For weights that have zeros in their entries, one can add a constant to the weight vector to avoid the problem of zero minimum entry of $\mathbf{w}$.

In the following, we illustrate via an example the values of the hyperparameters after convergence of SBL and WSBL. In the example, matrix $\mathbf{A}$ is constructed to have entries taken from zero mean and unit variance Gaussian distribution. The vector $\mathbf{x}$ has three randomly selected non-zero entries, all equal to 4. Additive white Gaussian noise is added to $\mathbf{Ax}$ with 10 dB SNR. The weight vector $\mathbf{w}$ is constructed by assigning the value of 0.1 or 1 corresponding to zero and non-zero values of $\mathbf{x}$. Fig. 3.1 (a) shows the SBL hyperparameters (in logarithmic scale) after convergence. One can see from the figure that the hyperparameters can take any value, and there is theoretically no upper bound on how large each α_i is. There is also high variance among the α_is with high values, corresponding to zero x_is. Fig. 3.1 (b) shows the hyperparameters of WSBL after convergence. One can see that in this case, the are either concentrated around $1/w_{is}^2$ (equivalently, the corresponding x_is are 0), or around 0 (equivalently, the corresponding x_is are nonzero). This behavior makes picking a threshold to differentiate between zero and nonzero values of α_is much easier than in SBL. In the next section, we discuss the using of MUSIC estimate as a weight vector, and provide an approach for setting the value of the threshold η.

3.3.2 Selection of MUSIC as a Weight

In the previous subsection, we mentioned that a coarse estimate of the underlying vector can be used as a weight vector. In this subsection, we study the use of MUSIC estimate as the weight vector. In general, the MUSIC estimate of $\mathbf{x}$ contains large values corresponding to the support of $\mathbf{x}$ and small values elsewhere. In the following, we study the behavior of the estimate outside the support of $\mathbf{x}$, and also provide a method to determine the threshold η in (3.14).

MUSIC [56] is a subspace estimation method, initially proposed for Direction Of Arrival (DOA) estimation in sensor array processing. Recently, MUSIC has been used for sparse vector support estimation [85] in the context of the problem of (3.1). In

particular, sparse vector support recovery can be seen as a special case of the DOA problem with discretized DOA angle space [85]. Given a set of snapshot observations, MUSIC estimates the signal subspace using eigendecomposition of the covariance matrix of the observations. The eigenvectors corresponding to the smallest $m-k$ eigenvalues represent the noise subspace, where k is the sparsity level of the sparse vector. Since the columns of $\mathbf{A}$ that correspond to the support of $\mathbf{x}$ span the same subspace as the signal subspace of the covariance matrix, their projections to the noise subspace will be almost zero. Based on that idea, the MUSIC estimate of x_i is defined as

$$\hat{x}_i = \frac{1}{\|\mathbf{E}_N \mathbf{a}_i\|_2^2} = \frac{1}{\|(\mathbf{I} - \mathbf{A}_s(\mathbf{A}_s^T\mathbf{A}_s)^{-1}\mathbf{A}_s^T)\mathbf{a}_i\|_2^2}, \tag{3.16}$$

where $i = 1, 2, ..., m$, $\mathbf{E}_N$ represents the noise subspace, $\mathbf{A}_s$ represents a matrix with normalized columns from $\mathbf{A}$ that belong to the support of $\mathbf{x}$, and $\mathbf{a}_i$ is i^{th} unit length column of $\mathbf{A}$. Therefore, on plotting the $\hat{x}_i$ versus i, a peak at $i = m$ will indicate that $\hat{x}_m$ belongs to the support of $\mathbf{x}$.

For a Gaussian matrix $\mathbf{A}$, the statistics of the distribution of the denominator in (3.16) are stated in the following proposition.

Proposition 2. *Consider (3.1), and assume that the entries of* $\mathbf{A}$ *are distributed as* $N(0, \frac{1}{m})$. *Let* $\mathbf{A}_s$ *be a matrix containing the columns of* $\mathbf{A}$ *that correspond to the support of* $\mathbf{x}$, *and let* k *be the sparsity level of* $\mathbf{x}$. *Define the approximate distance between the signal subspace and the column* $\mathbf{a}_i$ *as*

$$t \triangleq \|(\mathbf{I} - \mathbf{A}_s\mathbf{A}_s^T)\mathbf{a}_i\|_2^2, \tag{3.17}$$

where $\mathbf{a}_i$ *is a column of* $\mathbf{A}$ *that does not correspond to the support of* $\mathbf{x}$. *The random variable* t *has the following mean and variance:*

$$\mathbf{E}\{t\} = \frac{k + m^2 - mk + k^2}{m^2} \approx 1 - \frac{k}{m}. \tag{3.18}$$

$$\mathbf{Var}\{t\} = 2m\left(\frac{k + m^2 - mk + k^2}{m^3}\right)^2. \tag{3.19}$$

Proof. See Appendix A. □

The above proposition indicates that when m is large, the approximate distance for all $\mathbf{a}_i$s that do not belong to the support of $\mathbf{x}$ has low variance. This allows one to approximate $\hat{x}_i$ in (3.16) as the reciprocal of the expected value of t. Also, the approximate distance between the columns $\mathbf{a}_i$ that do not correspond to the support of the sparse vector and the signal subspace has values around 1, indicating that $\hat{x}_i$s for those columns are around 1. Also, one can easily show that the approximate distance is small (around 0) for all $\mathbf{a}_i$s that belong to the support of $\mathbf{x}$, which corresponds to large values of $\hat{x}_i$ for those columns. This suggests that MUSIC is a good weight vector for the proposed WSBL. Further, it results in no weight entries that have values close to zero.

The expected value and the variance of t along with Chebyshevs inequality can be used to set the threshold η. From Chebyshevs inequality [86] we have

$$p(|t - (1 - \frac{k}{m})| \geq p_i) \leq \frac{s^2}{p_i^2}, \tag{3.20}$$

where s is the variance as defined in (3.19), and p_i is a scalar. Eq. (3.20) can be decomposed as the sum of two probabilities

$$p(t \geq 1 - \frac{k}{m} + p_i) + p(t \leq 1 - \frac{k}{m} - p_i) \leq \frac{s^2}{p_i^2}. \tag{3.21}$$

Since we are interested in the second probability in (3.21) as it represents a smaller distance to the signal subspace (or a larger response in MUSIC), we can ignore the first term in (3.21) to get an upper bound of probability of interest, and rewrite (3.21) as

$$p(t \leq 1 - \frac{k}{m} - p_i) \leq \frac{s^2}{p_i^2} = p_{th}. \tag{3.22}$$

On setting p_{th} to a specific probability, we can find the value of p_i, which can then be used to find an upper limit for η as

$$\frac{1 + 2(1 - \frac{k}{m} - p_i)}{\mu_i^2 + \Sigma_{ii} + 2/(1 - \frac{k}{m} - p_i)} > \eta. \tag{3.23}$$

Since μ_i^2 and Σ_{ii} are both positive, we can drop them to get an upper limit of the threshold η, as follows

$$\frac{1 + 2(1 - \frac{k}{m} - p_i)}{2/(1 - \frac{k}{m} - p_i)} = \eta_{max} > \eta. \tag{3.24}$$

Equation (3.24) gives an upper limit for η such that with probability p_{th}, the distance between a column that does not belong to the support of the underlying signal and the signal subspace is less than $1 - \frac{k}{m} - p_i$.

One can see that calculating η_{max} requires the sparsity level. However, η_{max} does not change significantly for different values of k when k is small. For instance, when $m = 64$ and $p_{th} = 0.004$, the difference between η_{max} for $k = 1$ and $k = 10$ is small (around 0.0574). Here, we choose the threshold based on our experience with simulations, i.e., we take it to the average of η_{max} for different values of k, multiplied by a scalar that is less than one. One can choose that threshold to maximize the probability of detection, or minimizing the probability of false alarm. At this point we do not have a way to compute the optimal threshold, and hope to address this issue in our future work

Using the above observation, WSBL algorithm can be summarized as shown in Algorithm 1.

Algorithm 1: WSBL Estimation

Input : The weight vector $\mathbf{w}$, m, and p_{th}
Output: The estimated vector $\mathbf{x}$

1 Calculate η_{av} as the average of η_{max} for different values of k
2 Set $\eta = 0.9\eta_{av}$
3 Calculate $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ using (3.7) and (3.8), respectively
4 Calculate α_i and σ^2 using (3.13) and (3.11), respectively. If the convergence criterion is satisfied, go to step 5, otherwise go to step 3
5 Set $\mathbf{x} = \boldsymbol{\mu}$

3.4 Extension to the complex case

In many signal processing scenarios, the use of complex dictionary matrices, responses, and signals is crucial. in this section, we extended the proposed approach to accommodate with complex numbers scenarios. To apply WSBL to the case of complex numbers, one can parameterize complex entries systems in (3.1) into a real entries system as follows [87, 88]

$$\hat{\mathbf{y}} = \hat{\mathbf{A}}\hat{\mathbf{x}} + \hat{\mathbf{n}}, \tag{3.25}$$

where

$$\hat{\mathbf{y}} = \begin{bmatrix} \Re(\mathbf{y}) \\ \Im(\mathbf{y}) \end{bmatrix}, \hat{\mathbf{n}} = \begin{bmatrix} \Re(\mathbf{n}) \\ \Im(\mathbf{n}) \end{bmatrix}, \hat{\mathbf{x}} = \begin{bmatrix} \Re(\mathbf{x}) \\ \Im(\mathbf{x}) \end{bmatrix}, \\ \hat{\mathbf{A}} = \begin{bmatrix} \Re(\mathbf{A}) & -\Im(\mathbf{A}) \\ \Im(\mathbf{A}) & \Re(\mathbf{A}) \end{bmatrix}, \tag{3.26}$$

where $\hat{\mathbf{y}} \in R^{2m}$, $\hat{\mathbf{n}} \in R^{2m}$, $\hat{\mathbf{A}} \in R^{2m \times 2n}$, $\hat{\mathbf{x}} \in R^{2n}$, $\Re(x)$ represents the real part of complex number x, and $\Im(x)$ represents the imaginary part of a complex number x. Using the above representation, and following the same approach as we did in the real case, the update rules in this case is similar to the rules for the real case.

One should note that since the complex number system is parametarized into a real system, all the aforementioned propositions are still valid for complex number systems.

3.5 SBL at low Signal to Noise Ratio scenarios

In this section we study the SBL performance under low signal to noise ratio environment.

The SBL basis selection is based on choosing hyperparameters that maximize the probability of $\mathbf{y}$ being observed; $\mathbf{y}$ is an m-dimensional vector, following a Gaussian model [26]. To study the SBL performance under low SNR, we view the observation vector as a Gaussian process, i.e., $p(\mathbf{y}) = \mathcal{N}(0, \mathbf{C})$, where $\mathbf{C}$ can be expressed as [26]

$$\mathbf{C} = \sigma^2 \mathbf{I} + \sum_{i=1}^{n} \alpha_i^{-1} \mathbf{a}_i \mathbf{a}_i^T. \tag{3.27}$$

Changing α_i affects the size and the shape of the Gaussian distribution, while changing σ affects only the size of that distribution.

In SBL, σ^2 and α_i are estimated so that they make the observed vector most probable. This corresponds to aligning the Gaussian distribution to the observation vector, giving higher probability of selection (assigning low values) to those vectors $\mathbf{a}_i$s that are aligned to the output vector. On the other hand, $\mathbf{a}_i$s that are not well aligned to the observation vector will have lower probabilities to be selected as descriptor basis for the observation vector. This can be clearly observed when the estimated sparse vector

$\boldsymbol{\mu}$ is written as

$$\boldsymbol{\mu} = \left[\mathbf{I} - \sigma^{-2}\mathbf{F}^{-1}\mathbf{A}^T\mathbf{A}(\mathbf{I} + \sigma^{-2}\mathbf{F}^{-1}\mathbf{A}^T\mathbf{A})^{-1}\right]\sigma^{-2}\mathbf{F}^{-1}\mathbf{A}^T\mathbf{y}. \tag{3.28}$$

In (3.28), the term inside the square brackets, depends on the mutual coherence of the dictionary matrix $\mathbf{A}^T\mathbf{A}$ and the priors $\mathbf{F}^{-1}$. The remaining terms in (3.28) depend on the priors and the correlation between the observed vector and the columns of $\mathbf{A}$. If we assign equal prior probabilities, the most correlated columns in $\mathbf{A}$ will have higher probabilities to be selected as non-zero entries in the final estimation. This points to a problem for the case of low signal to noise ratio, as in that case, the deviation from the actual noise free observation may be large such that the actual vector may not be the most correlated with the observation. The above discussion is illustrated in Fig. 3.2 (a), with sensing matrix $\mathbf{A} = \begin{bmatrix} 1 & 1 & 0 & -1 \\ 0 & 1 & 1 & 1 \end{bmatrix}$, $\mathbf{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix}^T$, and SNR equal to 20 dB. The red circle represents the actual noise free observation, while the black circle represents the observation after additive noise. Each arrow represents one column from the sensing matrix $\mathbf{A}$. One can see that the two dimensional Gaussian distribution is oriented such that it is aligned to the observation vector, and in this case, it gives large contribution to the second column from the matrix $\mathbf{A}$ by assigning low value to the corresponding α_i. The estimated vector after convergence is $\begin{bmatrix} 0.21 & 0.85 & 0 & 0 \end{bmatrix}^T$. Fig. 3.2 (b) shows the case for the same scenario, but SNR equal to 5 dB. As expected, when the SNR is low, the observation vector (black circle) deviates from the noise free observation (red circle). Aligning the Gaussian distribution in the direction of the noisy observation will not reflect the actual solution. In this case, the solution gives to the first column in $\mathbf{A}$ the largest contribution for describing the observation, and to the second column a small contribution (note that the second column represents the actual solution). The estimated sparse vector after convergence is $\begin{bmatrix} 1.19 & 0.17 & 0 & 0 \end{bmatrix}^T$.

For the weighted approach, we assign different parameters a_i and b_i to the Gamma distribution that describes the priors of α_i. To understand the effect of assigning different parameters, we need to introduce the variational approximation to SBL, which is discussed in the next Section.

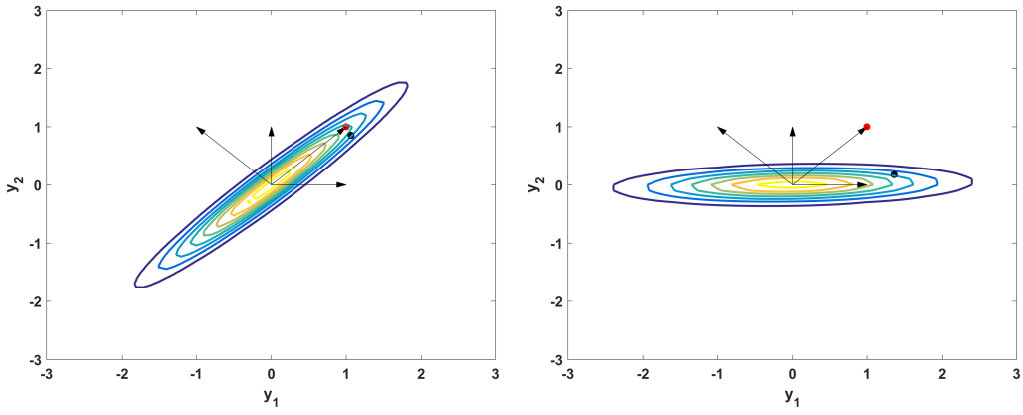


Figure 3.2: Contour of Gaussian distribution $p(\mathbf{y})$ after convergence in SBL for (a) SNR = 20 dB, and (b) SNR = 5. The red circle represents the actual noise free observation, while the black circle represents the observation after additive noise.

3.6 Variational Approximation of $p(\mathbf{x})$

In [19], it was shown that the SBL is a variational approximation, which selects the model that is the most probable to describe the observed vector. Also, the analysis of why we get a sparse solution from such approximation was also discussed in [19]. In this section, we use variational approximation theory to express $p(x_i)$ in its dual form, and introduce variational parameters for the proposed WSBL, which will help us explain the substantial improvement of WSBL as compared to SBL, especially at low SNRs.

The general framework of variational approximation relies on representing a convex function in its dual form [19]. Suppose that $f(x)$ is a convex function, the dual representation of $f(x)$ is

$$f(x) = \sup_{\lambda}(\lambda x - f^*(\lambda)), \tag{3.29}$$

where $f^*(\lambda)$ is the conjugate function of $f(x)$, defined as

$$f^*(\lambda) = \max_{x}(\lambda x - f(x)). \tag{3.30}$$

Eq. (3.29) can be written as

$$f(x) \geq \lambda x - f^*(\lambda), \tag{3.31}$$

for all λ. Eq. (3.31) represents a lower bound for $f(x)$.

On setting $g_i = \alpha_i^{-1}$, $p(x_i)$ can be written as

$$p(x_i) = \int p(x_i \mid g_i)p(g_i)dg_i = C(b + \frac{x_i^2}{2})^{-(a+0.5)}, \tag{3.32}$$

with $C = \frac{b^a \Gamma(a+0.5)}{\sqrt{2\pi}\Gamma(a)}$. One can see that as a and b approach zero, the pdf in (3.32) encourages sparsity due to its small tail and sharp peak at the origin. In (3.32), substituting x_i^2 with v, and taking the log, we get (for intermediate steps, please see Appendix E)

$$f(v) = \log(C) - (a + .5)\log(b + \frac{v}{2}), \tag{3.33}$$

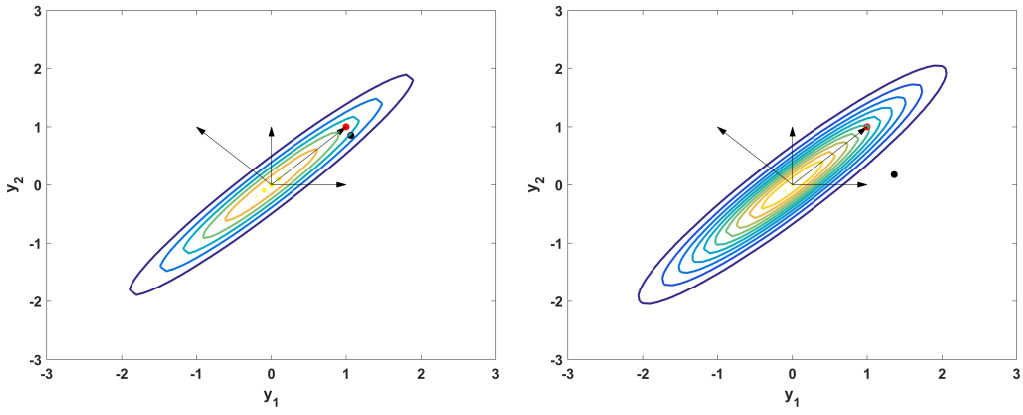
where $f(v) = \log(p(v))$, which is convex in v. Taking the dual of (3.33), substituting back v with x_i^2, and taking the exponent, we get

$$p(x_i) \geq \frac{2^{3\,a-1}\, b^a\, \Gamma\left(a + \frac{1}{2}\right)^2 \mathrm{e}^{-\frac{w^2+2\,b-g-2\,a\,g}{2\,g}}}{g^{a+\frac{1}{2}}\, \pi\, \Gamma\left(2\,a\right)\, \left(2\,a+1\right)^{a+\frac{1}{2}}}. \tag{3.34}$$

Eq. (3.34) holds with equality when $g_i = \frac{x_i^2+2b}{1+2a}$. This can be easily checked by differentiating w.r.t. g_i, and equating to zero. In that case, the corresponding x_i equals

$$x_i = \pm\sqrt{g_i(1 + 2a) - 2b}. \tag{3.35}$$

In SBL, if we take $a = b \approx 0$, we can rewrite (3.35) as $x_i = \pm\sqrt{g_i}$, which gives no limit to how small g_i can be. On the other hand, when we assign $a_i = 1/w_i$, and $b_i = w_i$, (3.35) can be written as $x_i = \pm\sqrt{g_i(1 + 2a_i) - 2b_i)}$. Since we expect x_i to be real, the quantity under the square root should be non negative, which corresponds to $g_i \geq \frac{2b_i}{1+2a_i}$, from which we have a lower limit for g_i. In WSBL, large w_i corresponds to small a_i and large b_i. In this case, $g_i \geq \frac{2b_i}{1+2a_i} \approx 2b_i = 2w_i$, which forces g_i to be large when w_i is large. This effectively increases the contribution of the corresponding column in the matrix $\mathbf{A}$ in (3.27), even when there is a large deviation from the actual noise free observation due to low SNR; and therefore, increasing the performance of WSBL under low SNR scenarios. The above discussion is shown in Figs. 3.3 (a) and (b) for the same scenario as described in Section slowromancapiii@. One can see that the Gaussian distributions are oriented towards the noise free observation (red circles) for both 20 dB and 5 dB SNRs, due to the high contribution of the second column of



(a) (b)

Figure 3.3: Contour of Gaussian distribution $p(\mathbf{y})$ after convergence in WSBL for (a) SNR = 20 dB, and (b) SNR = 5. The red circle represents the actual noise free observation, while the black circle represents the observation after additive noise.

the matrix $\mathbf{A}$. The high contribution of the second column of $\mathbf{A}$ is due to the lower bound of the corresponding hyperparameter g_2. On the other hand, when w_i is small, one can see that $\alpha_i = \frac{1}{g_i} \leq \frac{1+2{w_i}^{-1}}{2w_i} \approx \frac{1}{w_i^2}$, which gives an upper limit of the value of α_i when w_i is small. This upper limit simplifies the selection of the threshold that distinguishes between zero and non-zero entries in $\mathbf{x}$, as discussed previously in Section slowromancapiii@.

Figs. 3.4 and 3.6 show $p(x_i)$s (red curve) and its variational approximation (green curve) when $a = b \approx 0$, after convergence, for the examples presented in Section slowromancapiii@, and for SNR = 20, and 5 dB, respectively. One can see that $p(x_i)$ represents an upper limit of the variational approximation. From $p(x_i)s$, it is clear that all the entries have the same distributions, which indicates that they have the same chance to be non-zero in the final estimation.

Figs. 3.5 and 3.7 show the pdf $p(x_i)$s (red curve) and its variational approximation (green curve) when $a = w_i^{-1}$ and $b = w_i$, after convergence, with SNR = 20 and 5 dB, respectively. One can see that $p(x_i)$ represents an upper limit of the variational approximation. Due to the fact that the choice of parameters a_i and b_i differ in each distribution, there is a different upper limit for each α_i, which is small (corresponds to non-zero x_i) when w_i is large. Also, from $p(x_i)$s, one can see that $p(x_2)$ has a wider range, which suggests that there is a higher probability for the second entry to be

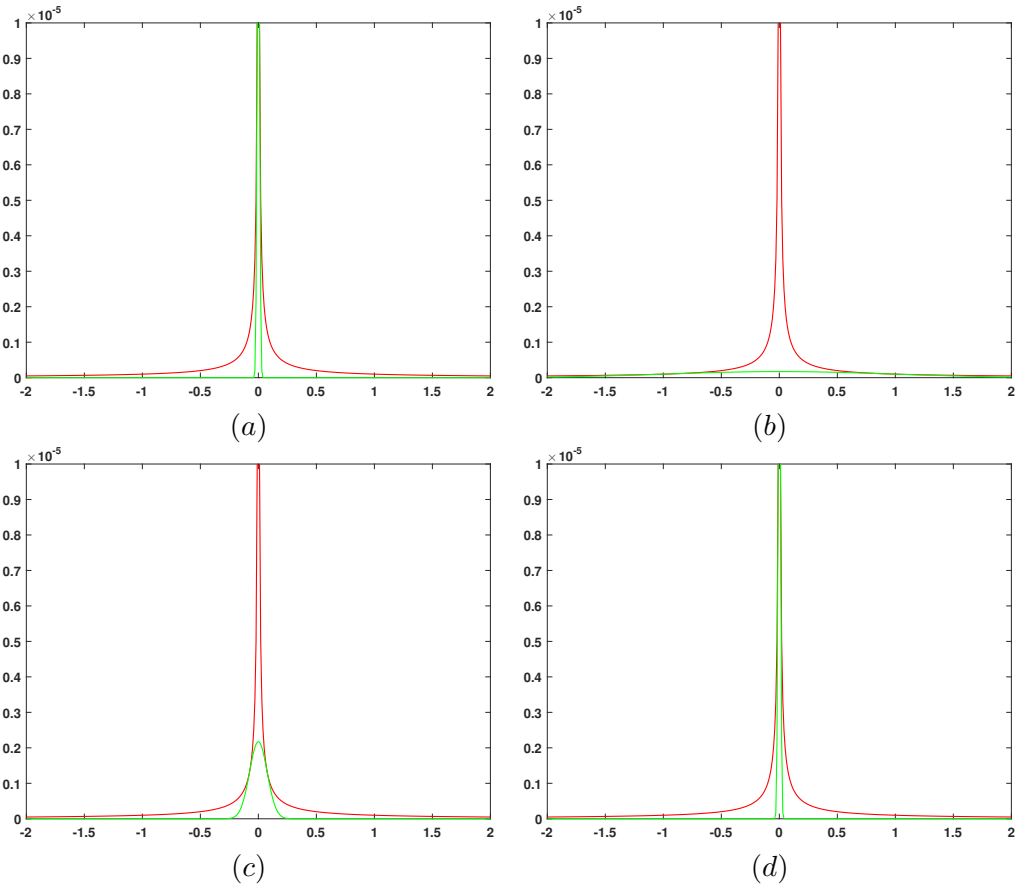


Figure 3.4: Variational Approximation of the example in Section slowromancapiii@ with SNR= 20 dB after convergence, when $a = b \approx 0$. Plots (a), (b), (c), and (d) represent the distribution of $p(x_i)$ (red curve), and its variational approximation (green curve) for $i = 1, 2, 3,$ and 4, respectively.

non-zero as compared to the other entries.

To summarize, WSBL outperforms SBL in low SNR scenarios due to the effect of the weights, which give more importance to some hyperparameters, by making them more probable to be selected as basis in describing the observations.

3.7 Simulation Results

In this section, we present simulation results for the proposed WSBL, and compare its performance with SBL, Sparse Bayesian Support knowledge (BSN) that is proposed in [29], and MSBL that is proposed in [30]. BSN uses a partly erroneous knowledge of the support to estimate the sparse vector, while MSBL uses multiple snapshots to improve the final estimation of the sparse vector, assuming stationary sources across all

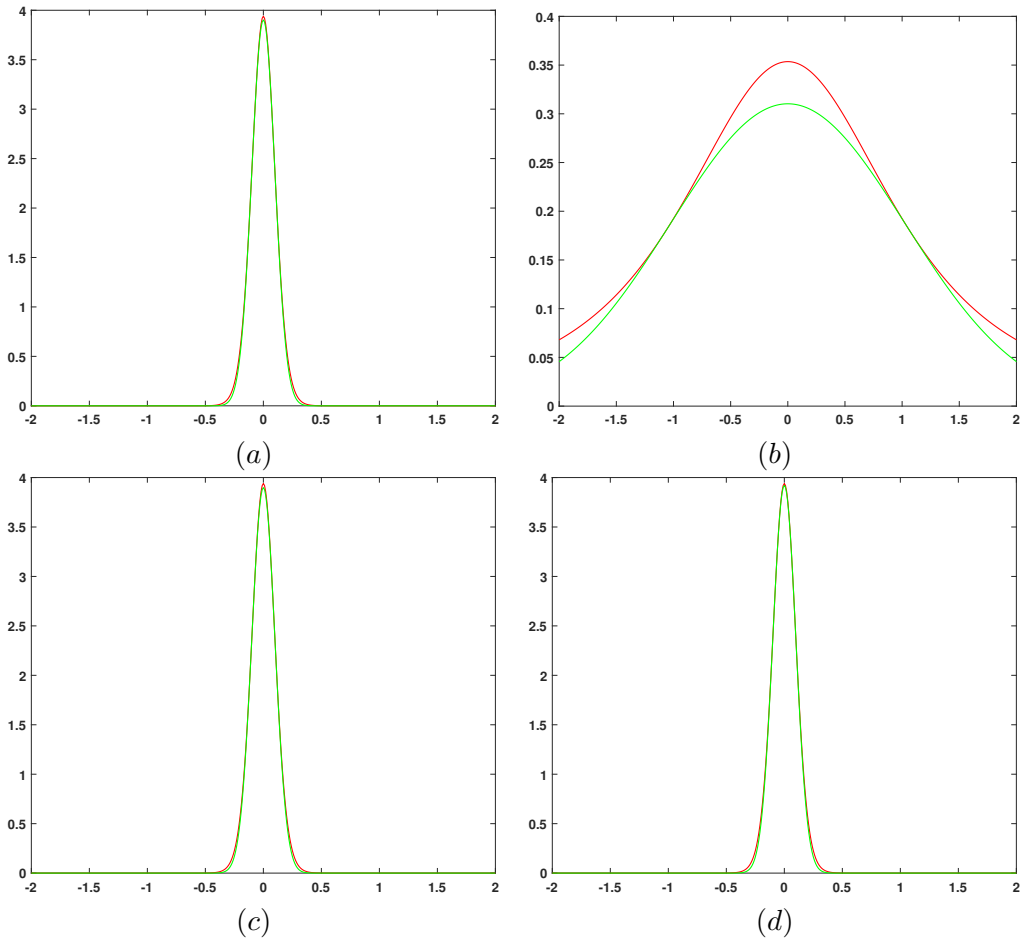


Figure 3.5: Variational Approximation of the example in Section slowromancapiii@ with SNR= 20 dB after convergence, when $a_i = w_i^{-1}$, and $b_i = w_i$, where $w = [0.1\ 1\ 0.1\ 0.1]^T$. Plots (a), (b), (c), and (d) represent the distribution of $p(x_i)$ (red curve), and its variational approximation (green curve) for $i = 1, 2, 3,$ and 4, respectively.

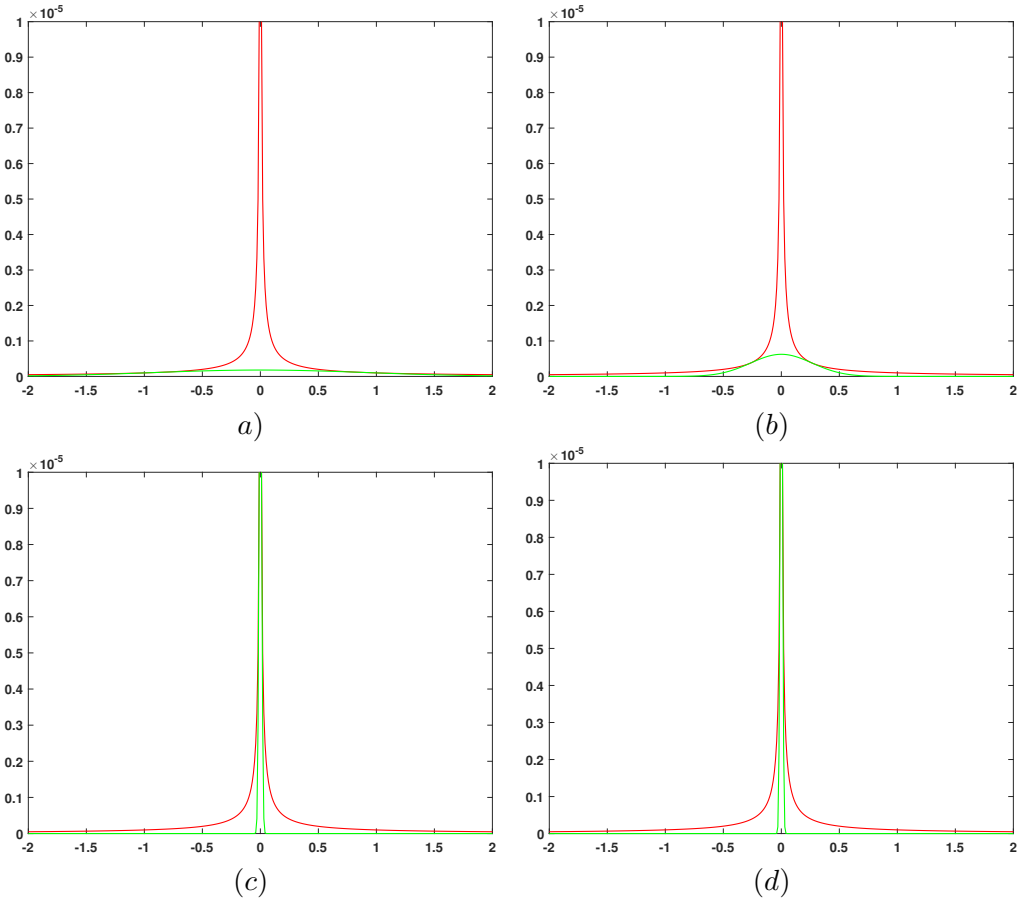


Figure 3.6: Variational Approximation of the example in Section slowromancapiii@ with SNR= 5 dB after convergence, when $a = b \approx 0$. Plots (a), (b), (c), and (d) represent the distribution of $p(x_i)$ (red curve), and its variational approximation (green curve) for $i = 1, 2, 3,$ and 4, respectively.

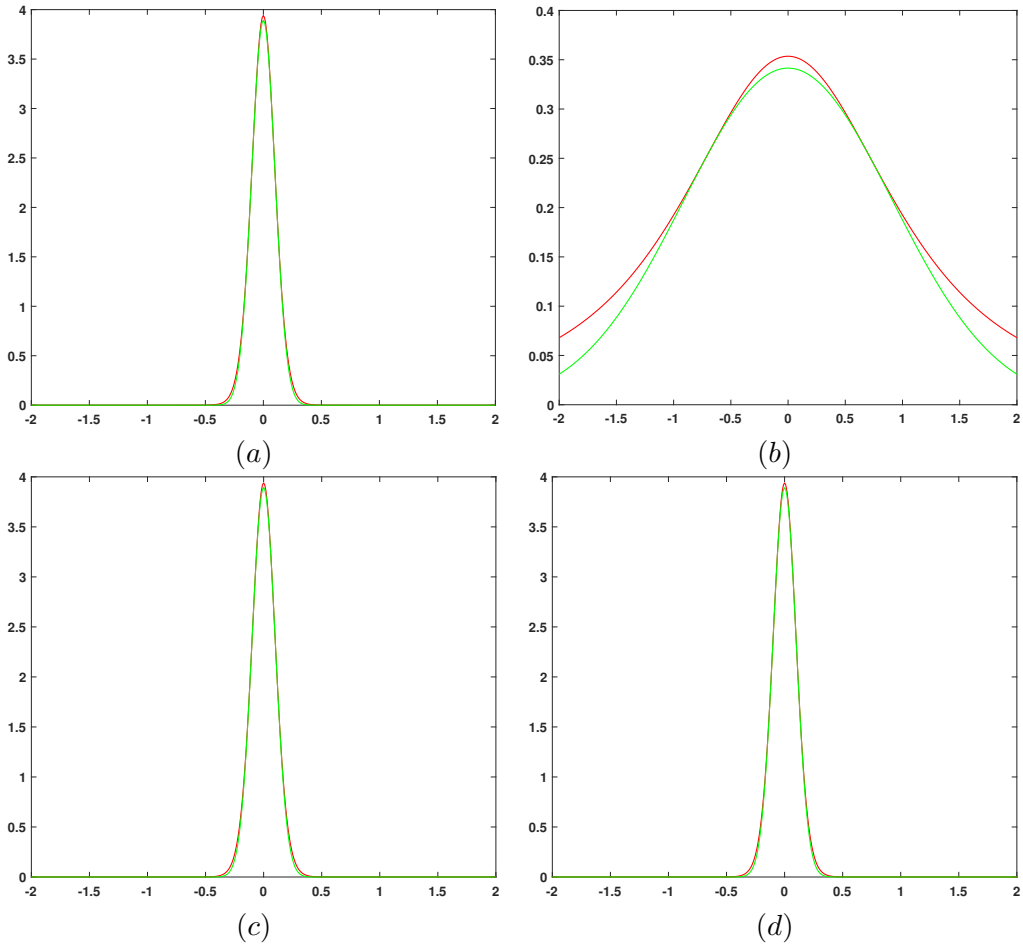


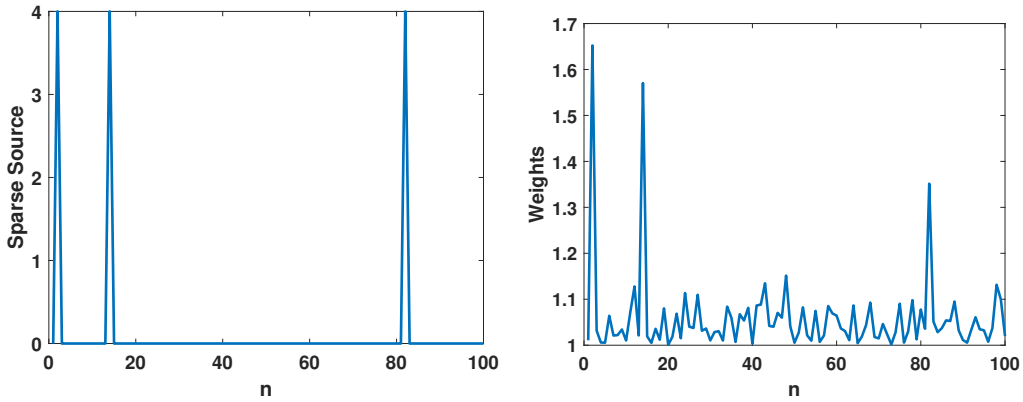
Figure 3.7: Variational Approximation of the example in Section slowromancapiii@ with SNR= 5 dB after convergence, when $a_i = w_i^{-1}$, and $b_i = w_i$, where $w = [0.1\ 1\ 0.1\ 0.1]^T$. Plots (a), (b), (c), and (d) represent the distribution of $p(x_i)$ (red curve), and its variational approximation (green curve) for $i = 1, 2, 3,$ and 4, respectively.

snapshots. Two applications will be presented in this section, sparse vector recovery for Gaussian dictionary matrices, and MIMO radar Direction Of Arrival (DOA) estimation.

3.7.1 Sparse vector recovery for Gaussian Dictionary Matrices

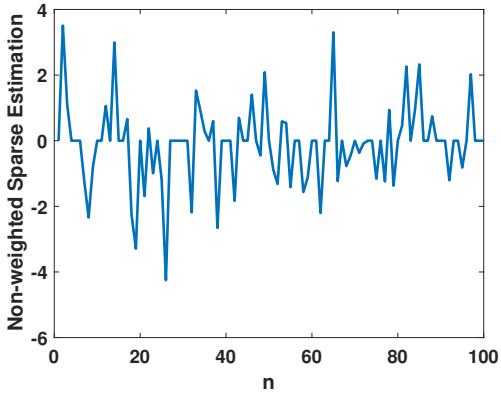
In the first simulation, a dictionary matrix of size 64×100 was constructed with entries following Gaussian distribution with zero mean and unit variance. A sparse vector with sparsity $k = 3$ was constructed, and the observations were obtained according to (3.1) with SNR of 0 dB. The weights were constructed using the MUSIC solution (see eq. (3.16)), which assumed the availability of 5 snapshots, and that the sources were in the same positions across snapshot. For BSN, we set the support to be the set of entries in the weight vector that have values 5% bigger than the reciprocal of (3.18); this simulates on average an erroneous support set with cardinality of 20. The MSBL estimation was based on the snapshots that were used for MUSIC estimation. Fig. 3.8 shows the actual sparse signal, the weights, and the final estimation results by SBL, BSN, MSBL, and WSBL. One can see that WSBL successfully estimates the location of the non-zero elements, while SBL and BSN do not perform well at such low SNR. While MSBL retrieves the actual sources, it also yields some false sources. It can be observed that the weights (MUSIC estimate) for most of the entries that do not belong to the support of the sparse signal have values that are close to $\frac{1}{1-\frac{k}{m}} = 1.05$, which supports our analysis of Section slowromancapiii@ (i.e., the reciprocal of (3.18)) .

Monte Carlo simulations with 1000 trials were also performed. In each trial, a dictionary matrix $\mathbf{A}$ of size 64×100 was constructed with entries following a zero mean and unit variance Gaussian distribution. k indices were randomly selected as the support of the sparse vector $\mathbf{x}$; the corresponding nonzero values were taken from a Gaussian distribution of mean 5 and standard deviation of 0.25. Observations were obtained according to (3.1) with SNR of 5 dB, and 0 dB. We exclude simulations for higher SNRs, due to the similar performances among different approaches. In each trial, the MUSIC solution based on 10 snapshots was used as the weight vector in WSBL. The Receiver Operating Characteristics (ROC) curve, average energy leakage, and mean squared error were used to compare performances among MUSIC, SBL, BSN, MSBL,

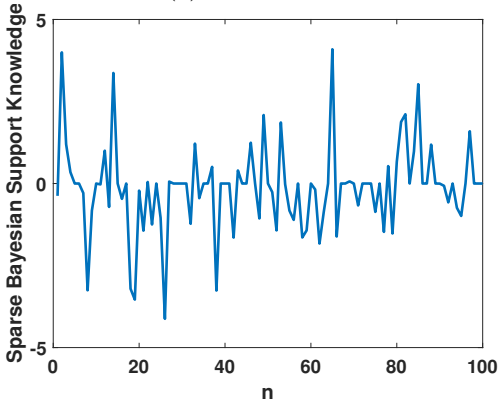


(a) The sparse vector

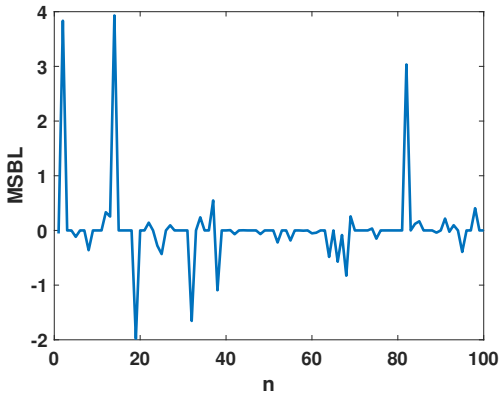
(b) The weights



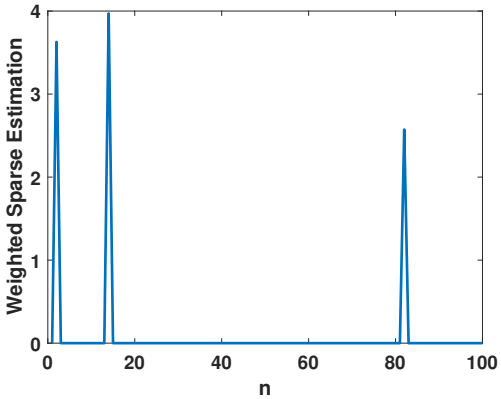
(c) The SBL estimate



(d) The SBN estimate



(e) The MSBL estimate



(f) The WSBL estimate

Figure 3.8: Single trial simulation with SNR= 0 dB after convergence. (a) The actual sparse vector. (b) Weights used in WSBL. (c) SBL estimate. (d) SBN estimate. (e) MSBL estimate. (f) WSBL estimate.

and WSBL. For ROC, successful estimation was declared if all the non-zero entries were successfully estimated. The average energy leakage was used to measure sources activity outside the true support of the sparse vector, while mean squared error was used to indicate how far the estimated sparse vector from the real one. Energy leakage was calculated by finding the second norm of the entries of the estimated vector that do not belong to the true support of the underlying sparse vector as follows. Let S be the set of true support indices of the sparse vector and $\bar{S}$ its complement. Let $\hat{\mathbf{x}}^i$ and $\mathbf{x}^i$ be the estimated and real sparse vectors at Monte Carlo iteration i, respectively. For M Monte Carlo trails, average leakage energy is defined as

$$E_{\text{leakage}} = \frac{1}{M}\sum_{i=1}^{M} \|\hat{\mathbf{x}}^i_{\bar{S}}\|_2, \tag{3.36}$$

while average mean square error is defined as

$$E_{\text{error}} = \frac{1}{M}\sum_{i=1}^{M} \|\mathbf{x}^i - \hat{\mathbf{x}}^i\|_2^2. \tag{3.37}$$

The average energy leakage provides an indication of the existence of false sources, and their strength in the estimated vector. For WSBL, the threshold η was set to be slightly less than η_{max} corresponding to $p_{th} = 0.004$ (see eq. (3.24)), while for SBL η was set to 1000. For BSN, the support was estimated as the case of single trial simulation discussed above. Figs. 3.9, 3.10, and 3.11 show the ROC curves for MUSIC, SBL, BSN, and WSBL for $k = 3, 4$ and 5 sources, at 5 dB, and 0 dB SNR, using Algorithm 1. One can see that, while SBL and BSN perform well in high and medium SNR, their performance degrades rapidly as the SNR decreases. WSBL and MSBL show identical performance under different scenarios. It can also be observed that WSBL clearly improves on the MUSIC estimate (used also as the weight vector), and exhibits robustness at low SNR. Increasing the number of snapshots would probably improve the MUSIC estimate, however, in a nonstationary signal scenario it may not be possible to use many snapshots, as will be shown in the third set of simulations below.

Also, it can be observed from Fig. 3.12 that WSBL exhibit the least average leakage energy among all approaches for different SNRs and different sparsity levels. Table slowromancapi@ shows the mean square error of WSBL, SBL, BSN, and MSBL for

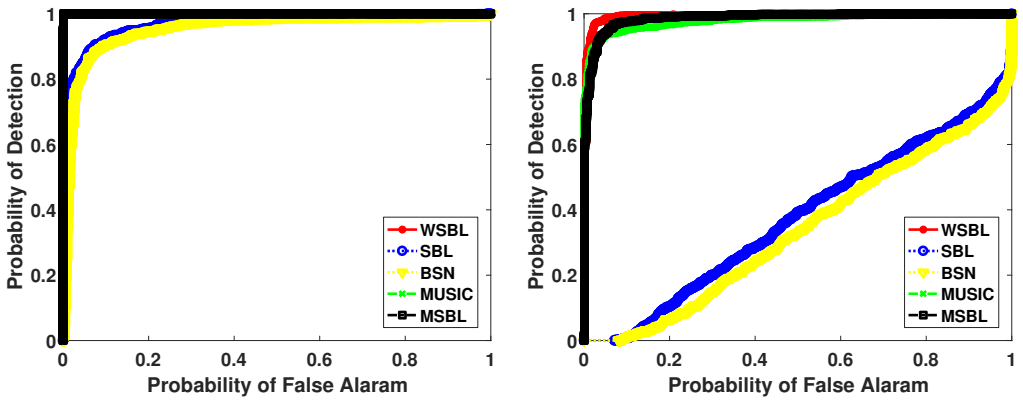


Figure 3.9: ROC curves for sparsity level of $k = 3$, and a) SNR=5 dB, b) SNR=0 dB.

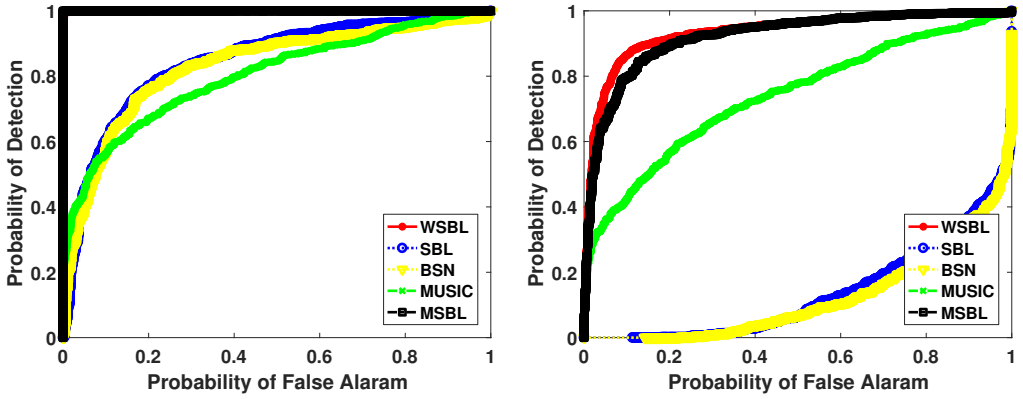


Figure 3.10: ROC curves for sparsity level of $k = 4$, and a) SNR=5 dB, b) SNR=0 dB.

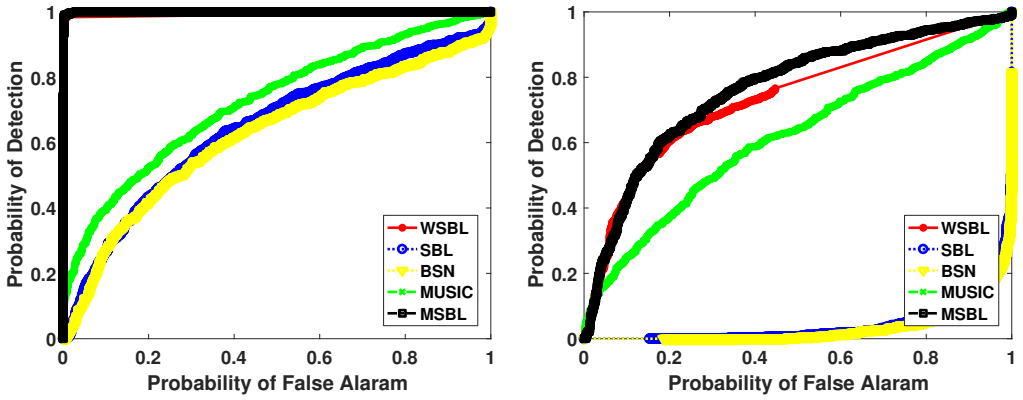


Figure 3.11: ROC curves for sparsity level of $k = 5$, and a) SNR=5 dB, b) SNR=0 dB.

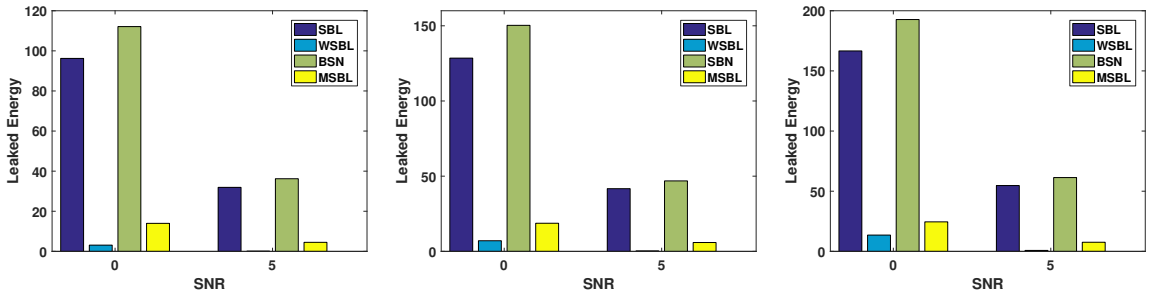


Figure 3.12: Leakage Energy for different SNRs, and a) $k = 3$, b) $k = 4$, c) $k = 5$.

different sparsity levels and different SNRs. One can see that WSBL shows the least mean square error among all the other approaches for all scenarios.

In the third simulations set, we consider the same simulation settings as the second simulations set, except we consider non-stationary sources. The sparse sources were generated such that they changed their support in each snapshot. The performance for all approaches was evaluated by comparing the estimated and the actual sparse vector based on the first snapshot. Figs. 3.13, and 3.14 show the ROC for different number of sources with different SNRs. As expected, the performance of SBL and BSN does not change, as their estimations are obtained based on a single snapshot. On the other hand, we see that the performance of MUSIC is degraded rapidly, due to the non-stationary sources. While both MSBL and WSBL degrade as the SNR decreases, WSBL appears to be more robust. Table slowromancapii@ shows the mean square error for WSBL, SBL, SBN, and MSBL, for different sparsity level and different SNRs. As expected, the mean squared errors for SBL and BSN are similar to the errors in Table slowromancapi@. We also see an increase in the error for both WSBL and MSBL due to the non-stationarity of the sources, and we notice that WSBL shows less error as compared to MSBL. Also, Fig. 3.15 shows that WSBL has the least average leakage energy as compared to all approaches for different scenarios.

We also compare the complexity of each approach in terms of the average time required for convergence. Table slowromancapiii@ shows the average time over all Monte Carlo simulations that were performed as discussed above. As expected, MSBL involved higher complexity, as it processes all snapshots as a matrix of size $n \times M_{snap}$

Table 3.1: Mean-square Error for SBL, SBN, MSBL and WSBL for Different Scenarios, 1000 Monte Carlo Trials

	WSBL			SBL			BSN			MSBL		
SNR / k	10	5	0	10	5	0	10	5	0	10	5	0
3	0.2	0.66	3.26	10.82	35.62	102.68	12.12	39.25	118.32	1.66	5.12	16.44
4	0.37	1.35	8.29	14.95	47.26	145.46	16.63	51.64	168.06	2.38	7.53	24.03
5	0.62	2.56	17.39	20.29	63.63	187.98	21.62	69.41	217.51	3.29	10.6	33.53

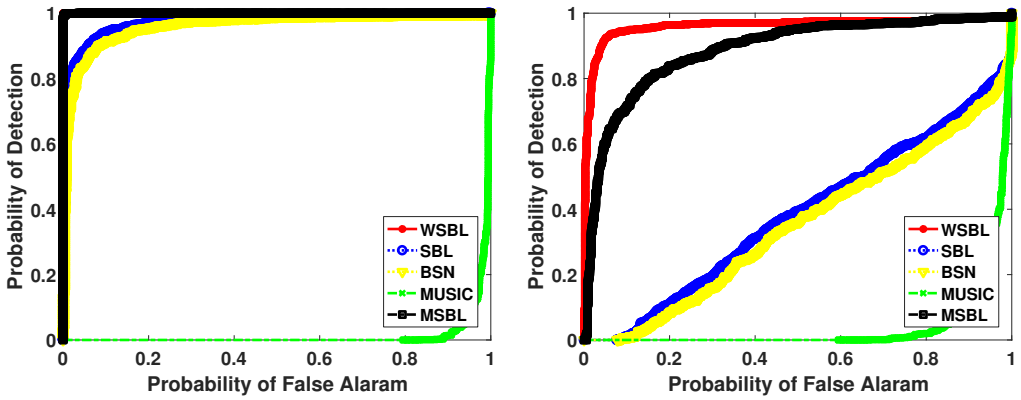


Figure 3.13: ROC curves for sparsity level of $k = 3$ non-stationary sources, and a) SNR=5 dB, b) SNR=0 dB.

to find the final estimate, where M_{snap} is the number of snapshots. WSBL provides the least complexity among all the approaches. Both WSBL and SBL involve the same complexity in the beginning of the iteration, while the complexity of each iteration reduces as column are excluded from the estimation. For WSBL, the number of excluded columns is larger due to the upper limit on the hyperparameters, which leads to faster convergence, and thus lower overall complexity.

Table 3.2: Mean-square Error for SBL,SBN, MSBL and WSBL for non-stationary sources, 1000 Monte Carlo Trials

	WSBL			SBL			BSN			MSBL		
SNR / k	10	5	0	10	5	0	10	5	0	10	5	0
3	0.31	1.15	7.87	10.78	35.47	106.4	12.1	37.75	122.06	3.04	8.91	26.45
4	0.61	2.45	17.91	15.15	47.7	145.47	16.36	52.68	167.55	5.34	14.55	42.93

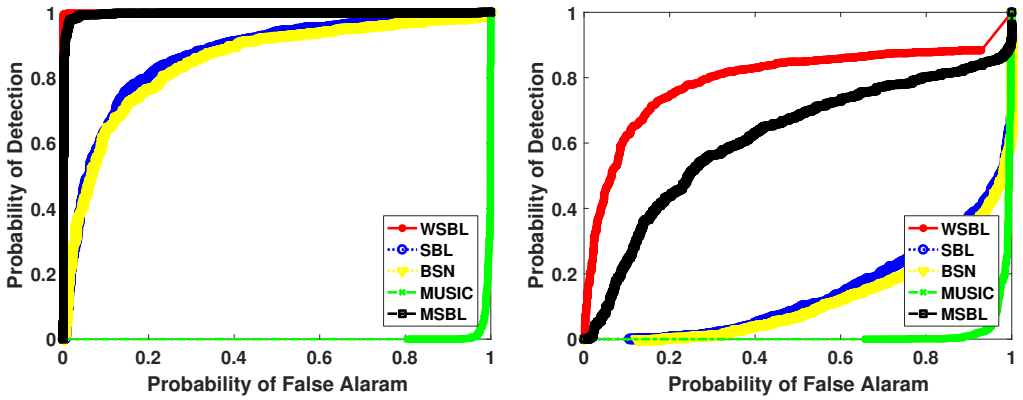


Figure 3.14: ROC curves for sparsity level of $k = 4$ non-stationary sources, and a) SNR=5 dB, b) SNR=0 dB.

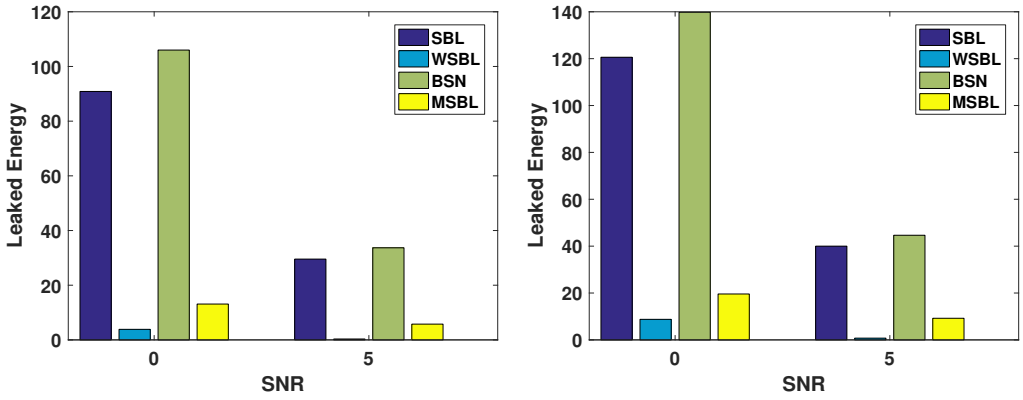


Figure 3.15: Leakage Energy for different SNRs, and a) $k = 3$, b) $k = 4$.

Table 3.3: Average time of convergence in seconds

WSBL	SBL	BSN	MSBL
0.001039394	0.04345625	0.082087879	1.225569697

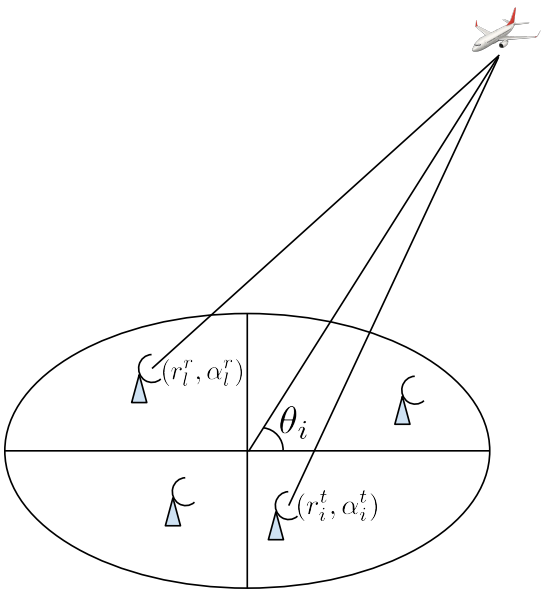


Figure 3.16: Radar geometry

3.7.2 MIMO Radar DOA Estimation

We now consider the problem of estimating DOA of colocated MIMO radar setup. First, we discuss the signal model of MIMO radar that we used in this chapter, and then present the simulation results.

MIMO radar signal model

We follow the signal model for MIMO radar that is described in [6], and is also shown in Fig. 3.16. Let N_t, and N_r represent the number of transmitters and receivers, respectively. The transmitters and receivers are randomly distributed within a circle of radius r. Let θ_i represent the azimuth angle of target i, the polar coordinates of the l^{th} receiver/transmitter is $r_l^{r/t}, \alpha_l^{r/t}$ with respect to the circle origin of transmitters/receivers. Also, let us define $\mathbf{v}(\theta_i)$ as

$$\mathbf{v}(\theta_i) = [e^{j2\pi/\lambda \eta_1^t(\theta_i)} \dots e^{j2\pi/\lambda \eta_{N_t}^t(\theta_i)}], \tag{3.38}$$

where λ represents the carrier wavelength, and $\eta_l^{r/t}$ is the projected distance of the l^{th} transmitters/receiver on the θ_i axis, i.e.,

$$\eta_l^{r/t}(\theta_i) = r_l^{r/t} \cos(\theta_i - \alpha_l^{r/t}). \tag{3.39}$$

On the transmitter side, define the matrix $\mathbf{S}$ to be the transmitted samples from all transmitters, i.e., $\mathbf{S} = [\mathbf{s}_1[0] \dots \mathbf{s}_{N_t}[(L-1)]]$, where $\mathbf{s}_i[m]$ is a signal of length L that is transmitted from i^{th} transmitter. Dividing the DOA space into different grid points, the received signal at the l^{th} receiver can be described as [6]

$$\mathbf{z}_l = \mathbf{\Psi}_l \mathbf{x} + \mathbf{n}_l, \tag{3.40}$$

where

$$\mathbf{\Psi}_l = [e^{j2\pi/\lambda \eta_l^r(\theta_1)} \mathbf{S}\mathbf{v}(\theta_1) \dots e^{j2\pi/\lambda \eta_l^r(\theta_n)} \mathbf{S}\mathbf{v}(\theta_n)], \tag{3.41}$$

where θ_i corresponds to the i^{th} grid point, and $\mathbf{x}$ is a vector of length n. Each entry in $\mathbf{x}$ corresponds to the existence or the absence of a target at a specific grid point. Assuming small number of targets exist in the scene of interest, a small subset of samples at each receiver can be used to estimate targets' DOAs. One way to obtain such subset is by taking random projections of the received samples at each receiver, i.e.,

$$\mathbf{r}_l = \mathbf{\Phi}_l \boldsymbol{z}_l = \mathbf{\Phi}_l \mathbf{\Psi}_l \mathbf{x} + \tilde{\mathbf{n}}_l, \tag{3.42}$$

with $\mathbf{\Phi}$ represent a zero-mean Gaussian random matrix with unit variance. Each receiver then forward these samples to the fusion center which combines the outputs from N_r receivers, and formulates the problem as

$$\mathbf{r} = [\mathbf{r}_1^T \dots \mathbf{r}_{N_r}^T]^T = \mathbf{A}\mathbf{x} + \mathbf{n}, \tag{3.43}$$

where $\mathbf{A} = [(\mathbf{\Phi}_1 \mathbf{\Psi}_1)^T \dots (\mathbf{\Phi}_n \mathbf{\Psi}_n)^T]^T$.

Simulation results

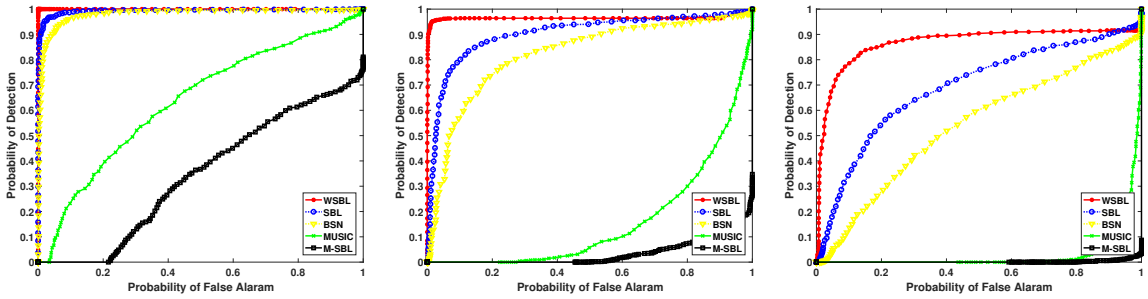
Here, we present simulation results for the proposed WSBL, and compare its performance with SBL, BSN, MUSIC, and MSBL for a colocated MIMO radar scenario. Here, we only include low SNR scenario, because of the comparable performances among different approaches in higher SNR scenarios.

We simulated a MIMO radar which contains number of transmitter/receiver of 10 and 7, respectively. The carrier frequency was set to 5 GHz, the transmitters and receivers were distributed uniformly at random inside a circle of radius 10 m, and

the transmitters sent 10 orthogonal waveforms. The distance between the center of the transmitter/receiver region and the target scene was 10 Km. In this simulation, uniformly discretized angle space that ranges between -90^{o} to 90^{o} was considered with 1^{o} step. Random projections from each receiver were taken by multiplying the received samples with a zero mean, unit variance random Gaussian matrix of size (15×512). These sets of samples were used to estimate DOA of the targets. We compared the performance of the proposed approach with MUSIC, SBL, BSN, and MSBL

Monte Carlo simulations with 1000 trials were performed for the MIMO system described above. In each trial, k targets were randomly distributed around the scene of interest. At the receivers, the reflected signals due to these targets were collected, and additive white Gaussian noise was added to the received signals with SNR equals to 0 dB. MUSIC was used to estimate the target angles. For MUSIC, the snapshots matrix was constructed based on the received signals from N_r receivers, i.e., 7 snapshots were used for MUSIC estimation. The estimation from MUSIC was also used as a weight vector for the proposed WSBL. For BSN, The erroneous support setting is similar to the case of Gaussian matrix simulations. The snapshots that were used for MUSIC were also used for MSBL.

Fig. 3.17 shows the ROC curves for MUSIC, SBL, WSBL, BSN, and MSBL, for $k = 2, 3$ and 4 sources, and 0 dB SNR. One can see that the performances of SBL and BSN degrade rapidly with the increase of the number of targets in such low SNR. On the other hand, one can easily notice the inferior performance of MUSIC and MSBL. The poor performances of MSBL and MUSIC can be explained as follows. Due to the compression performed, the snapshots used in both MUSIC and MSBL are effectively the received signals multiplied by the compression matrix. As multiple snapshots are required by both MUSIC and MSBL, and due to the different compression of each snapshot, each snapshot has a different steering vector. Thus, effectively, the targets appear as non-stationary for both MUSIC and MSBL. This explains the poor performances of MSBL and MUSIC in the MIMO radar scenario. The low performance of MSBL is because MSBL misses one of the actual targets and/or detects false targets. This increases the probability of false alarm, and decreases the probability of detection.



(a) (b) (c)

Figure 3.17: ROC curves for SNR = 0 dB and a) $k = 2$,b) $k = 3$, c) $k = 4$ for colocated MIMO radar with 10 transmitters and 7 receivers.

This dramatically affects the performance of MSBL. MUSIC on the other hand is not a sparse recovery method, so even if MUSIC misses some targets, the probability of detection can not be zero. This explains why there is a slight increase in the performance of MUSIC as compared to MSBL. Also, WSBL shows robustness with the increase of the number of sources, as compared to SBL and BSN. Although MUSIC provide as rather poor estimate of the solution, it provides a good enough weight for WSBL to achieve significantly better estimation performance than SBL and BSN.

3.8 Conclusion

We have proposed the Weighted Sparse Bayesian Learning approach which relies on a single snapshot for sparse signal recovery. In WSBL, the hyper-parameters have a distinct set of parameters, which introduces more degrees of freedom of this optimization and thus improves the accuracy. An approximation of the underlying sparse vector, obtained for example via a method that does not encourage sparsity, can be used to determine the relative importance of the hyperparameters. Theoretical and simulation results have shown that by exploiting available coarse estimate of the sparse vector results in significant improvement in probability of detection, and probability of false alarm, especially at low SNR scenarios, as compared to SBL and BSN. The proposed approach also exhibits the least overall complexity in terms of convergence time among all the other approaches. Also, we have shown that the MUSIC estimate can be used effectively as a weight vector. In contrast to MSBL, the proposed approach has shown more robustness in estimating non-stationary sources under low SNR environments

for limited number of sources. WSBL has also demonstrated the least average leakage energy among all the other approaches in all scenarios. WSBL has also shown robustness in estimating DOA of targets in MIMO radar under low SNR as the number of targets increases.

Chapter 4

Bernoulli Sparse Bayesian Learning for Basis Selection

4.1 Introduction

As shown in the previous chapters, the weighted approach invokes substantial improvement over the non-weighted approach. MUSIC was used as a weight vector in the previous works. However, multiple snapshots requirement in MUSIC may impose a problems in systems when stationary sources across snapshots is not guaranteed. Here, we propose to model the rate parameters as Bernoulli random variables. The Bernoulli parameters are estimated based on the observations. The parameters that are used (along with the observations) to estimate the Bernoulli parameters are estimated offline using machine learning approach. We name the proposed approach as Bernoulli Sparse Bayesian Learning (BSBL). BSBL assigns a low or high value to the rate parameter of the Gamma pdf corresponding to each element of the sparse vector; a large rate parameter makes the Gamma pdf more concentrated around zero, and correspondingly, the entry of the sparse vector non-zero. Distinct rate parameter assignment guides BSBL to choose the true support of the corresponding sparse vector.

The chapter is organized as follows. Section 4.2 presents the proposed Bernoulli Sparse Bayesian Learning approach, Section 4.3 discusses an approach to find the Bernoulli random variable parameters for the proposed BSBL approach. Section 4.4 presents simulation results, while Section 4.5 provides concluding remarks.

4.2 The Proposed BSBL Approach

Consider the linear system

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n} \tag{4.1}$$

where $\mathbf{n} \in \mathbf{R}^m$ is white Gaussian noise with zero mean and variance σ^2, $\mathbf{A} \in \mathbf{R}^{m \times n}$ represents a dictionary matrix, $\mathbf{x} \in \mathbf{R}^n$ is a sparse vector, and $\mathbf{y} \in \mathbf{R}^m$ represents the observation vector. The probability density function (pdf) of $\mathbf{y}$ given $\mathbf{x}$ and σ^2 equals

$$p(\mathbf{y} \mid \mathbf{x}, \sigma^2) = (2\pi\sigma^2)^{-m/2} \exp(-\frac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2). \tag{4.2}$$

Assuming that the entries in $\mathbf{x}$ are identically independent Gaussian distributed with zero mean, and the variance of i^{th} entry in $\mathbf{x}$, x_i, is α_i^{-1}, the conditional pdf of $\mathbf{x}$ given $\boldsymbol{\alpha}$ is [26]

$$p(\mathbf{x} \mid \boldsymbol{\alpha}) = \prod_{i=1}^{n} \left(\frac{\alpha_i}{2\pi}\right)^{1/2} \exp\left(\frac{-x_i^2 \alpha_i}{2}\right), \tag{4.3}$$

where $\boldsymbol{\alpha} = \begin{bmatrix} \alpha_1 & \alpha_2 & ... & \alpha_n \end{bmatrix}^T$ is the vector of hyperparameters. For a large α_i, the corresponding x_i has small variance, and since the mean of x_i is zero, x_i has high probability of being zero. In SBL [26], the entries of $\boldsymbol{\alpha}$ are assumed to be independent Gamma-distributed with parameters a (shape) and b (rate). The parameter σ^{-2} is also assumed to follow a Gamma distribution with parameters d and e, which are usually set to zero [26]. In SBL, all α_is are assumed to be identically distributed with parameters a, b, which are fixed to very small values [26]. The performance of SBL degrades in low SNR scenarios [89].

In WSBL [89], each α_i is assumed to be distributed with different parameters, a_i, b_i; those parameters are computed based a coarse estimate of the sparse vector. While improved performance as compared to SBL was reported in low SNR scenarios [89], obtaining a coarse estimate involves additional complexity.

Different than SBL and WSBL, here we assume that we have knowledge of some prior probability that the x_i is non-zero. This guides BSBL to choose the true support of the sparse vector. Estimation of these probabilities depends on a set of parameters that can be estimated offline using an empirical approach. Given the set of estimated parameters, prior probabilities can be obtained efficiently. Estimating the set of parameters and prior probabilities will be discussed in section slowromancapiii@. Using estimated prior probabilities, BSBL assigns low or high values to the rate parameters of the $\boldsymbol{\alpha}$ random variables. The higher the rate parameter of the Gamma distributed random variable, the more the pdf is concentrated around zero, which corresponds to a

higher probability that the corresponding x_i is non-zero. In a nutshell, BSBL chooses the most probable hyperparameters $(\boldsymbol{\alpha}, \mathbf{b}, \sigma^2)$ that best describe the observation vector and the prior probabilities.

Let us assume that a vector $\mathbf{p} = \begin{bmatrix} p_1 & p_2 & \dots & p_n \end{bmatrix}^T$ is available, where p_i is the probability that x_i is non-zero. Using such information, we treat the rate parameter, b_i, of each α_i as Bernoulli random variable, taking values c or 0 with probability p_i and $1 - p_i$, respectively, where c is some positive scalar. The rate random variable will be estimated along with the hyperparameters. We name this approach Bernoulli Sparse Bayesian Learning (BSBL). One should note that the higher the rate parameter b the more concentrated the Gamma pdf is around zero, which makes the corresponding α_i most probably zero, or equivalently, the corresponding x_i non-zero. As we will discuss in the next section, the required parameters used to estimate the vector $\mathbf{p}$ are estimated empirically offline, and the complexity associated with estimating $\mathbf{p}$ from these parameters is low.

Let us assume that the hyperparameters $\boldsymbol{\alpha}$ given $\mathbf{b}$ are still independent Gamma distributed, i.e.,

$$p(\boldsymbol{\alpha} \mid \mathbf{b}) = \prod_{i=1}^{n} \text{Gamma}(\alpha_i; a, b_i), \tag{4.4}$$

where b_is are independent Bernoulli random variables, and $\mathbf{b} = \begin{bmatrix} b_1 & b_2 & \dots b_n \end{bmatrix}^T$. The posterior pdf can be rewritten as

$$p(\mathbf{x}, \boldsymbol{\alpha}, \sigma^2, \mathbf{b} \mid \mathbf{y}) = p(\mathbf{x} \mid \mathbf{y}, \boldsymbol{\alpha}, \sigma^2) p(\boldsymbol{\alpha}, \sigma^2, \mathbf{b} \mid \mathbf{y}), \tag{4.5}$$

with $p(\mathbf{x} \mid \mathbf{y}, \boldsymbol{\alpha}, \sigma^2)$ following Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ [26], given as

$$\boldsymbol{\mu} = \sigma^{-2} \boldsymbol{\Sigma} \mathbf{A}^T \mathbf{y}, \tag{4.6}$$

$$\boldsymbol{\Sigma} = (\sigma^{-2} \mathbf{A}\mathbf{A}^T + \mathbf{F})^{-1}, \tag{4.7}$$

where $\mathbf{F} = \text{diag}(\alpha_1, \alpha_2, ..., \alpha_n)$. On approximating $p(\boldsymbol{\alpha}, \sigma^2, \mathbf{b} \mid \mathbf{y})$ as a delta function at its mode [26], maximizing (4.5) can be decomposed into two distinct maximization problems. In the first problem, we estimate $\boldsymbol{\alpha}, \mathbf{b}$ and σ^2 that maximize $p(\boldsymbol{\alpha}, \sigma^2, \mathbf{b} \mid \mathbf{y})$. The

estimated parameters from the first problem are then used to maximize $p(\mathbf{x} \mid \mathbf{y}, \boldsymbol{\alpha}, \sigma^2)$ using (4.6) and (4.7). For $p(\boldsymbol{\alpha}, \sigma^2, \mathbf{b} \mid \mathbf{y})$, it holds that

$$p(\boldsymbol{\alpha}, \sigma^2, \mathbf{b} \mid \mathbf{y}) \propto p(\mathbf{y} \mid \boldsymbol{\alpha}, \sigma^2) p(\boldsymbol{\alpha} \mid \mathbf{b}) p(\sigma^2) p(\mathbf{b}), \tag{4.8}$$

where $p(\mathbf{b}) = \prod_{i=1}^{n} p(b_i)$. Taking the log of (4.8), and ignoring terms independent of $\boldsymbol{\alpha}, \sigma$ and $\mathbf{b}$, we obtain the following objective function:

$$\begin{aligned} L = & -\frac{1}{2}\log|\sigma^2\mathbf{I} + \mathbf{A}\mathbf{F}^{-1}\mathbf{A}^T| - \frac{1}{2}\mathbf{y}^T(\sigma^2\mathbf{I} + \mathbf{A}\mathbf{F}^{-1}\mathbf{A}^T)^{-1}\mathbf{y} \\ & + a\sum_{i=1}^{n}\log b_i + (a-1)\sum_{i=1}^{n}\log\alpha_i - \sum_{i=1}^{n} b_i\alpha_i + \sum_{i=1}^{n}\frac{b_i}{c}\log(p_i) \\ & + \sum_{i=1}^{n}(1 - \frac{b_i}{c})\log(1-p_i) \end{aligned} \tag{4.9}$$

where $\mathbf{I}$ is the identity matrix of appropriate size. Note that $p(\sigma^2)$ has disappeared from the objective function above because of the zero values of the constants d and e.

An iterative approach can be invoked to find the values of $\boldsymbol{\alpha}, \sigma^2$ and $\mathbf{b}$ that maximize (4.9). For $\boldsymbol{\alpha}$, differentiating with respect to α_i, and equating to zero, we get

$$\alpha_i^{(\text{new})} = \frac{1 + 2a}{\mu_i^2 + \Sigma_{ii} + 2b_i} \tag{4.10}$$

where μ_i represents the i^{th} entry in $\boldsymbol{\mu}$, and Σ_{ii} represents the i^{th} diagonal entry in $\boldsymbol{\Sigma}$. Differentiating with respect to σ^2, and equating to zero, we get

$$(\sigma^2)^{(\text{new})} = \frac{\|\mathbf{y} - \mathbf{A}\boldsymbol{\mu}\|_2^2}{m - \sum_{i=1}^{n}\gamma_i}, \tag{4.11}$$

with $\gamma_i = 1 - \alpha_i\Sigma_{ii}$. For b_i, let us first define the loss function L_b associated with $\mathbf{b}$ by collecting all terms that have b_is in (4.9), i.e.,

$$L_b = \sum_{i=1}^{n} L_{b_i} = \sum_{i=1}^{n} -b_i\alpha_i + \frac{b_i}{c}\log(p_i) + (1 - \frac{b_i}{c})\log(1-p_i). \tag{4.12}$$

We need to choose a combination of b_is such that (4.12) is maximized, with each entry $b_i \in \{0, c\}$. Following the independence assumption among b_is, maximizing (4.12) can be recast as maximizing each L_{b_i}. In this sense, the values of b_is is chosen as follows

$$b_i = \begin{cases} 0 & \log(1-p_i) > \log(p_i) - c\alpha_i \\ c & \log(1-p_i) < \log(p_i) - c\alpha_i. \end{cases} \tag{4.13}$$

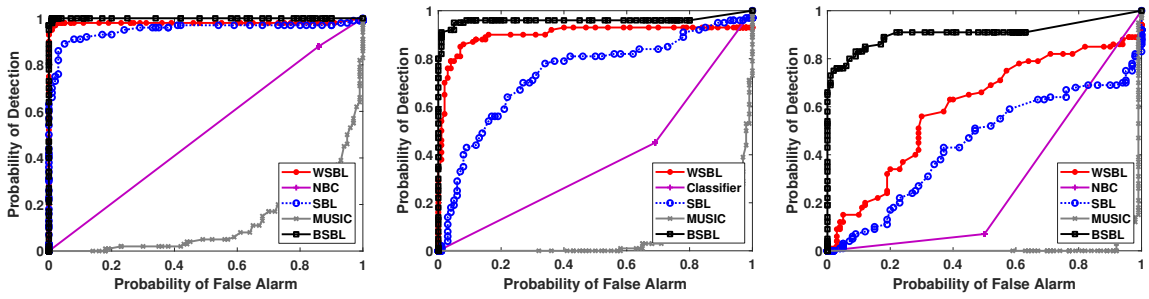
BSBL can be summarized as follows. First, estimate $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$, using (4.6), (4.7), respectively. Then update $\boldsymbol{\alpha}$, σ^2, and $\mathbf{b}$ using (4.10), (4.11), and (4.13). The aforementioned procedure is repeated until some convergence criterion is satisfied. After convergence, the estimated sparse vector is $\boldsymbol{\mu}$.

The above analysis considers the case of real dictionary matrix and observations. The above results can be easily extended to the complex number dictionary matrix and observations by following the same approach described in Section 3.4.

4.3 On estimating the prior probability vector p using training data

As mentioned in the previous section, the proposed BSBL needs the prior probabilities of each x_i to be active (non-zero). In this section, we provide a training method to estimate $\mathbf{p}$ based on an observation vector $\mathbf{y}$. We propose to use a Gaussian Naive Bayesian Classifier (NBC) [90] to obtain $\mathbf{p}$. The NBC classifier takes as feature vector the observation vector, and provides the probability distribution of the two classes "c" and "0". We implement n distinct classifiers, one for each entry of the sparse vector.

Before using NBC, we need to first train NBC using a training set. The training set can be constructed by generating M sparse vectors and their corresponding observations using (4.1), for the given sensing matrix, a specific sparsity level (k) and specific Signal to Noise Ratio (SNR). The M observations are then used to train the n NBC classifiers. Following the Independence assumption among observation vector entries assumed in NBC, and assuming that $p(y_k \mid b_i = c) = \mathcal{N}(\mu_k^{b_i=c}, \sigma_k^{b_i=c})$ and $p(y_k \mid b_i = 0) = \mathcal{N}(\mu_k^{b_i=0}, \sigma_k^{b_i=0})$, where $\mathcal{N}(\mu, \sigma)$ is the Gaussian kernel with μ mean and σ standard deviation, the M observations can be used offline to estimate the parameters $\mu_k^{b_i=c}, \mu_k^{b_i=0}, \sigma_k^{b_i=c}$, and $\sigma_k^{b_i=c}$. These parameters are then used to find the probability that each entry in $\mathbf{x}$ is active/non-active for any observation vector. The trained classifiers are then used to obtain $\mathbf{p}$ as follows. Given the observation $\mathbf{y}$, we obtain probability of each element to be active/non-active for each entry in $\mathbf{x}$ by applying $\mathbf{y}$ at the input of all n NBCs. $\mathbf{p}$ then represents the probabilities that each entry in $\mathbf{x}$ is active. The obtained probability vector $\mathbf{p}$ is then used in BSBL for the sparse estimation.



(a) (b) (c)

Figure 4.1: ROC curves for SNR = 0 dB, and a) $k = 3$, b) $k = 4$, c) $k = 5$.

While the NBC can also provide an estimate of the sparse vector $\mathbf{x}$, we here use only the probabilities of active classes. As it will be seen in the simulations Section, the BSBL estimate with the p produced by NBC is significantly better that the NBC estimate for the same number of observations.

4.4 Simulation Results

In this section, we present simulations results of the proposed BSBL applied to target angle estimation in a MIMO radar scenario, and comparisons to SBL [19], and WSBL with weights based on the MUSIC estimate [89]. The model described in Section 3.7 is used in the following simulations.

A MIMO radar with number of transmitters/receivers of 10 and 7, respectively, and with carrier frequency of 5 GHz was simulated. The transmitters and receivers were distributed uniformly within a circle of 10 m. The distance between the targets scene and the center of the circle containing the radar antennas was 10 Km. 10 Orthogonal waveforms from the transmitters were transmitted. The angle space considered in all simulations was between -90^o to 90^o, and the range was uniformly discretized with 0.667^o step. The number of samples received in each receiver was 512 samples. At each receiver, 15 random projections were extracted by multiplying the received samples with a random Gaussian matrix of mean 0 and unit valiance. These samples were then forwarded to the fusion center. We compared the performance of the proposed approach with SBL, and WSBL.

Monte Carlo simulations with 100 trials were performed for the MIMO radar described above. In each trial, k targets were randomly distributed in the angle space, and the reflections from these targets were collected at the radar receivers. Additive white Gaussian noise at SNR of 0 dB was added to the received samples at each receiver. We should note that for SNR greater than 0 dB the performance of all different approaches is similar. The MUSIC estimate used in WSBL was obtained based on multiple snapshots of the radar antennas, i.e., 15 snapshots. The priors and likelihood parameters that were used to obtain the probability vector $\mathbf{p}$ were estimated using $M = 2500$ observations, with sparsity level $k = 5$ and SNR $= 0$ dB. For BSBL, the positive scalar c was set to 0.1 in all simulations. We also tried different values for c without big difference in BSBL performance, which may indicate that the value of c has no big impact on the final estimation. Receiver Operating Characteristics (ROC) curves were used to compare the performances of BSBL, SBL, and WSBL. Detection was declared when all the targets were detected successfully.

Finding a high value for p_i gives us a good hint that x_i belongs to the support of the sparse vector. However, this by itself does not provide a good estimate of the sparse vector support. To illustrate this point, we constructed a sparse vector by assigning 1s to x_i's corresponding to $p_i > 0.5$, and setting the rest to zero. We refer to this approach as the NBC approach and use it to show how much BSBL improves over it.

Fig. 4.1 shows ROC curves for $k = 3, 4,$ and 5, and for SNR $= 0$ dB. One can see that SBL, WSBL, and BSBL result in comparable performance for $k = 3$. Also, there is a slight degradation of WSBL performance for $k = 4$ as compared to BSBL, while we see degraded performance of SBL. One can also see the substantial improvement in performance of BSBL for $k = 5$ case in comparison to SBL and WSBL. Also, one can see the improvement of BSBL over the NBC approach.

Although one would have done more training to improve the probabilities p_i, and thus improve the Classifier approach, this would increase the involved complexity associated with NBCs training. It appears that the proposed approach can make up for the lack of a large amount of training samples.

4.5 Conclusions

Bernoulli Sparse Bayesian Learning for basis selection has been proposed. The rate parameters of the hyperparameters are modeled as a Bernoulli random variables, and are estimated along with the hyperparameters. The parameters of the Bernoulli random variables are obtained based on the observation and also training data. The proposed approach exhibits improved performance as compared to SBL and WSBL for low SNR and different levels of sparisty.

Chapter 5

Conclusions and Future Works

In this dissertation, we have demonstrated the use of an estimate of the sparse vector to improve the sparse vector recovery. Also, we have demonstrated the use of machine learning approaches to effectively enhance sparse vector recovery. For the weighted ℓ_1-norm minimization problem, we have shown that the RSP conditions can be satisfied for the weighted approach by appropriately assigning weights, even if the original non-weighted approach does not satisfy the RSP conditions. Conditions on the weights have also been provided. Monte Carlo simulations have shown improved success rate as compared to widely used sparse vector recovery approaches. We also have evaluated the proposed approach performance in localizing active sources in an auditory experiment from EEG recordings. The proposed approach has successfully localized the active sources within the proximity of primary auditory cortex.

WSBL has also been proposed, in which the hyperparameters have a distinct set of parameters. These parameters are assigned using a coarse estimate of the underlying sparse vector. Using of MUSIC estimate with WSBL has been validated analytically, and we have shown through simulations the substantial improvement in ROC curves in low SNR cases as compared to SBL-based approaches. WSBL has been applied to compressed sensing MIMO radar DOA estimation, and has shown robustness in estimating DOA of targets under low SNR as the number of targets increases.

Bernoulli Sparse Bayesian Learning (BSBL) has also been proposed. In BSBL, the rate parameters of the hyperparameters are modelled as Bernoulli random variables, and are estimated with the hyperparameters. Machine learning approach has been used to estimate the parameters of Bernoulli random variables. We have applied BSBL on compressed sensing MIMO radar to estimate targets' DOAs. BSBL has shown enhanced

detection rate as compared to WSBL that uses MUSIC as weights in low SNR scenarios for different sparsity levels.

Initiated by the research proposed in this thesis, we would like also to provide a few future research directions as follows:

1. In Chapters 1 and 2, the use of MUSIC as a weight vector is demonstrated. The use of different estimations as weights is important to explore and can shed the light on the effects of different weights on the performance proposed approach in estimating the sparse solution. An example of such weights is the Spectral estimations, which have been used extensively in Phase Retrieval applications.

2. Regarding EEG source localization, we assume throughout the simulations that the active sources lie on the grid points. One approach is to consider the estimation of off-grid sources by using linear Taylor approximation of the off-grid sources using the neighboring grid points. This can find applications in BCIs, clinical neoroscience for epilepsy researches, and cognitive neuroscience to understand the brain functionality associated with complex tasks.

3. In WSBL and BSBL, we only consider on-grid targets only. However, in practice, the targets may not be exactly lie on one of the predefined grid points. One can consider expanding this approach to estimate off-grid targets' DOAs. This in turn increases the accuracy of the estimated targets' DOAs.

4. Fast SBL has been shown to have a lower complexity as compared to SBL. Along this, one could consider to extend WSBL and BSBL to Fast WSBL, and Fast BSBL, respectively. For this, one can consider a trade-off between the basis corresponds to the largest decrease in the loss functions of WSBL and BSBL, and the corresponding assigned weight in WSBL or the estimated Bernoulli parameters in BSBL. Decreasing complexity is important in applications where processing time is crucial, such Automotive Radar Systems.

5. In BSBL, we consider the use of NBCs to estimate the parameters of the Bernoulli

random variables. One can consider to deploy more sophisticated machine learning approaches and deep learning to improve the probability of detection rate for lower SNRs and larger number of targets. The trade off between the number of training samples, training time, and processing time should be considered in deciding the machine learning approach utilized with BSBL.

Appendix A

Proof of Theorem 2

To prove Theorem 2 we will prove its contrapositive, i.e., we will show that if a solution does not satisfy Theorem 1, then $\left|\mathbf{A}_{J0}^T(\mathbf{A}_{Js}^T)^\dagger \mathbf{u}_s\right| \succeq \mathbf{1}$.

Let $\mathbf{x}$ satisfy the RSP conditions (a),(b), and (c) except condition (d) of Theorem $1(i)$. On defining $\mathbf{v}$ such that $\mathbf{A^T v} = \mathbf{u}_s$, the aforementioned conditions can be rewritten as

$$\mathbf{A}_{J+}^T\mathbf{v} = \mathbf{1}, \mathbf{A}_{J-}^T\mathbf{v} = -\mathbf{1}, \left|\mathbf{A}_{J0}^T\mathbf{v}\right| \succeq \mathbf{1}. \tag{A.1}$$

Let us group together the first and second conditions in (A.1) to form the system $\mathbf{A}_{Js}^T\mathbf{v} = \mathbf{u}_s$, where $\mathbf{A}_{Js} = \begin{bmatrix} \mathbf{A}_{J+} & \mathbf{A}_{J-} \end{bmatrix}$, and $\mathbf{u}_s = \begin{bmatrix} \mathbf{1}^T & -\mathbf{1}^T \end{bmatrix}^T$. This is an underdetermined system, whose least square solution equals $\mathbf{v}_{ls} = (\mathbf{A}_{Js}^T)^\dagger \mathbf{u}_s$. Substituting $\mathbf{v}_{ls}$ in $\left|\mathbf{A}_{J0}^T\mathbf{v}\right| \succeq 1$, we get $|\mathbf{A}_{J0}^T(\mathbf{A}_{Js}^T)^\dagger \mathbf{u}_s| \succeq \mathbf{1}$, which completes the proof.

Appendix B

Proof of Theorem 3

First, we prove the necessary condition. Define $\mathbf{x}_p^\star$ and $\mathbf{x}_n^\star$ to be vectors containing the positive and negative entries of $\mathbf{x}^\star$ respectively, i.e.,

$$x_{pi}^\star = \begin{cases} x_i^\star & x_i^\star > 0 \\ 0 & x_i^\star \leq 0 \end{cases}, \text{ and } x_{ni}^\star = \begin{cases} -x_i^\star & x_i^\star < 0 \\ 0 & x_i^\star \geq 0 \end{cases}.$$

Then $\mathbf{x}^\star = \mathbf{x}_p^\star - \mathbf{x}_n^\star$ and $\|\mathbf{x}^\star\|_1 = \mathbf{1}^T\mathbf{x}_p^\star + \mathbf{1}^T\mathbf{x}_n^\star$. The problem of (2.18) can be rewritten as

$$\begin{aligned} \underset{\mathbf{x}_n, \mathbf{x}_p, \mathbf{v}, t}{\text{minimize}} \quad & h\mathbf{1}^T\mathbf{x}_p + h\mathbf{1}^T\mathbf{x}_n + t \\ \text{Subject to} \quad & \mathbf{x}_p \succeq \mathbf{0}, \mathbf{x}_n \succeq \mathbf{0} \\ & t \geq \|\mathbf{v}\|_2, \mathbf{v} = \mathbf{y} - \mathbf{A}\mathbf{x}_p + \mathbf{A}\mathbf{x}_n, \end{aligned} \tag{B.1}$$

with a solution $\mathbf{x}^\star = \mathbf{x}_p^\star - \mathbf{x}_n^\star$. The Lagrangian of (B.1) is

$$\begin{aligned} L = h\mathbf{1}^T\mathbf{x}_p + h\mathbf{1}^T\mathbf{x}_n + t - \boldsymbol{\lambda}_1^T\mathbf{x}_p - \boldsymbol{\lambda}_2^T\mathbf{x}_n \\ -\alpha(t - \|\mathbf{v}\|_2) + \boldsymbol{\lambda}_3^T(\mathbf{y} - \mathbf{A}\mathbf{x}_p + \mathbf{A}\mathbf{x}_n - \mathbf{v}), \end{aligned} \tag{B.2}$$

and the corresponding dual problem as defined in (2.19). On denoting with $\mathbf{v}^\star$, $\boldsymbol{\lambda}_1^\star$, $\boldsymbol{\lambda}_2^\star$, and $\boldsymbol{\lambda}_3^\star$ the solution to (2.19), and based on the strict complementary slackness, we have that $\lambda_{1i}^\star = 0$ when $x_{pi}^\star > 0$, and $\lambda_{2i}^\star = 0$ when $x_{ni}^\star > 0$. From the complementary slackness and the constraints (19.1) and (19.2) in (2.19), we get

$$\begin{aligned} \mathbf{a}_{pi}^T\boldsymbol{\lambda}_3^\star = h \quad \text{when} \quad x_i^\star > 0 \\ \mathbf{a}_{ni}^T\boldsymbol{\lambda}_3^\star = -h \quad \text{when} \quad x_i^\star < 0, \end{aligned} \tag{B.3}$$

where $\mathbf{a}_{pi}$ and $\mathbf{a}_{ni}$ are the columns of $\mathbf{A}$ associated with positive and negative elements of $\mathbf{x}^\star$, respectively.

By adding and subtracting (19.2) and (19.2) in (2.19), we get

$$\begin{aligned} \mathbf{A}^T\boldsymbol{\lambda}_3^\star &= \frac{\boldsymbol{\lambda}_2^\star - \boldsymbol{\lambda}_1^\star}{2} \quad (C3) \\ \boldsymbol{\lambda}_1^\star + \boldsymbol{\lambda}_2^\star &= 2h\mathbf{1}. \quad (C4) \end{aligned} \tag{B.4}$$

For zero $x_{pi}^\star$ and $x_{ni}^\star$, $\lambda_{1i}^\star$ and $\lambda_{2i}^\star$ are non-zero. From (C4) of (B.4), we have that $0 < \lambda_{1i}^\star < 2h$ and $0 < \lambda_{2i}^\star < 2h$, which implies that

$$-2h < \lambda_{2i}^\star - \lambda_{1i}^\star < 2h \quad \text{when} \quad x_i^\star = 0. \tag{B.5}$$

Combining (B.5) and (C3) in (B.4), we get

$$-h < \bar{\mathbf{a}}_i^T\boldsymbol{\lambda}_3 < h \quad \text{when} \quad x_i^\star = 0, \tag{B.6}$$

where $\bar{\mathbf{a}}_i$ is any column associated with zero entry in $\mathbf{x}^\star$.

For proving statement (d) in this theorem, we use Slater's condition, i.e.,

$$\mathbf{y}^T\boldsymbol{\lambda}_3^\star = h\mathbf{1}^T\mathbf{x}_p^\star + h\mathbf{1}^T\mathbf{x}_n^\star + \|\mathbf{v}^\star\|_2. \tag{B.7}$$

Since $\mathbf{y} = \mathbf{A}\mathbf{x}_p^\star - \mathbf{A}\mathbf{x}_n^\star + \mathbf{v}^\star$, we can rewrite (B.7) as follows

$$(\mathbf{A}^T\boldsymbol{\lambda}_3^\star - h\mathbf{1})^T\mathbf{x}_p^\star - (\mathbf{A}^T\boldsymbol{\lambda}_3^\star + h\mathbf{1})^T\mathbf{x}_n^\star + \boldsymbol{\lambda}_3^{\star T}\mathbf{v}^\star = \|\mathbf{v}^\star\|_2. \tag{B.8}$$

The first two terms in (B.8) are always zero. This is because, from (B.3), for non-zero $\mathbf{x}_p^\star$ and $\mathbf{x}_n^\star$, $\mathbf{A}^T\boldsymbol{\lambda}_3^\star - h\mathbf{1}$ and $\mathbf{A}^T\boldsymbol{\lambda}_3^\star + h\mathbf{1}$ are zeros. Also, these terms are zero when $\mathbf{x}_p^\star$ and $\mathbf{x}_n^\star$ are zeros. This implies that

$$\boldsymbol{\lambda}_3^{\star T}\mathbf{v}^\star = \|\mathbf{v}^\star\|_2 \Longrightarrow \boldsymbol{\lambda}_3^\star = \frac{\mathbf{v}^\star}{\|\mathbf{v}^\star\|_2} = \frac{\mathbf{y} - \mathbf{A}\mathbf{x}^\star}{\|\mathbf{y} - \mathbf{A}\mathbf{x}^\star\|_2}. \tag{B.9}$$

So, indeed, there is a $\mathbf{u} = \mathbf{A}^T\boldsymbol{\lambda}_3^\star \in R(\mathbf{A}^T)$ such that when $\mathbf{x}^\star$ is a solution to (2.18), we have that

$$\begin{cases} (a)\ u_i = h & \text{if } x_i^\star > 0 \\ (b)\ u_i = -h & \text{if } x_i^\star < 0 \\ (c)\ |u_i| < h & \text{if } x_i = 0 \end{cases}$$

(d) $\quad \mathbf{u} = \mathbf{A}^T\frac{\mathbf{y}-\mathbf{A}\mathbf{x}^\star}{\|\mathbf{y}-\mathbf{A}\mathbf{x}^\star\|_2}$.

Next, we will prove the sufficient condition, i.e., if for a given $\mathbf{x}^\star$, we have $\boldsymbol{\lambda}_3^\star$ such that

$$\boldsymbol{\lambda}_3^\star = \frac{\mathbf{y}-\mathbf{A}\mathbf{x}^\star}{\|\mathbf{y}-\mathbf{A}\mathbf{x}^\star\|_2}, \begin{cases} u_i = \mathbf{a}_i^T\boldsymbol{\lambda}_3^\star = h & \text{if } x^\star{}_i > 0 \\ u_i = \mathbf{a}_i^T\boldsymbol{\lambda}_3^\star = -h & \text{if } x^\star{}_i < 0 \\ |\mathbf{a}_i^T\boldsymbol{\lambda}_3^\star| < h & \text{if } x^\star{}_i = 0 \end{cases}, \tag{B.10}$$

then $\mathbf{x}^\star$ is a solution to (2.18). Assume that $\hat{\mathbf{x}} \neq \mathbf{x}^\star$ is the solution to (2.18). Then, it should hold that $h\|\hat{\mathbf{x}}\|_1 + \|\mathbf{y}-\mathbf{A}\hat{\mathbf{x}}\|_2 < h\|\mathbf{x}^\star\|_1 + \|\mathbf{y}-\mathbf{A}\mathbf{x}^\star\|_2$. Also, from the necessary condition of this theorem, we should have

$$\begin{aligned} &\mathbf{a}_i^T\hat{\boldsymbol{\lambda}}_3 = h && \text{if } \hat{x_i} > 0 \\ &\mathbf{a}_i^T\hat{\boldsymbol{\lambda}}_3 = -h && \text{if } \hat{x_i} < 0 \\ &|\mathbf{a}_i^T\hat{\boldsymbol{\lambda}}_3| < h && \text{if } \hat{x_i} = 0 \\ &\hat{\boldsymbol{\lambda}}_3 = \tfrac{\mathbf{y}-\mathbf{A}\hat{\mathbf{x}}}{\|\mathbf{y}-\mathbf{A}\hat{\mathbf{x}}\|_2}, \end{aligned} \tag{B.11}$$

Following the assumption of Slater's condition, and since the dual problem should attend its maximum at $\hat{\boldsymbol{\lambda}}_3$ and $\hat{\mathbf{v}} = \mathbf{y}-\mathbf{A}\hat{\mathbf{x}}$, we should have that

$$\begin{aligned} &\mathbf{y}^T\tfrac{\hat{\mathbf{v}}}{\|\hat{\mathbf{v}}\|_2} > \mathbf{y}^T\tfrac{\mathbf{v}^\star}{\|\mathbf{v}^\star\|_2} \\ &\Rightarrow (\mathbf{A}\hat{\mathbf{x}}+\hat{\mathbf{v}})^T\tfrac{\hat{\mathbf{v}}}{\|\hat{\mathbf{v}}\|_2} > (\mathbf{A}\mathbf{x}^\star+\mathbf{v}^\star)^T\tfrac{\mathbf{v}^\star}{\|\mathbf{v}^\star\|_2} \\ &\Rightarrow h\|\hat{\mathbf{x}}\|_1 + \|\mathbf{y}-\mathbf{A}\hat{\mathbf{x}}\|_2 > h\|\mathbf{x}^\star\|_1 + \|\mathbf{y}-\mathbf{A}\mathbf{x}^\star\|_2, \end{aligned} \tag{B.12}$$

which contradicts the first assumption requiring that $h\|\hat{\mathbf{x}}\|_1 + \|\mathbf{y}-\mathbf{A}\hat{\mathbf{x}}\|_2 < h\|\mathbf{x}^\star\|_1 + \|\mathbf{y}-\mathbf{A}\mathbf{x}^\star\|_2$. This implies that $\hat{\mathbf{x}}$ is not the solution to the problem of (2.18).

Appendix C

Proof of Theorem 4

From condition (c) of Theorem 3, if we choose w_{ii} such that $|w_{ii}\mathbf{a}_i^T\boldsymbol{\lambda}_3^\star| < h$, with $\boldsymbol{\lambda}_3^\star$ represents the solution of the dual variable of the problem define in (2.19), then $q_i = 0$. Starting from condition (c) of Theorem 3, we have that

$$\begin{aligned} & w_{ii}|\mathbf{a}_i^T\boldsymbol{\lambda}_3^\star| < h \\ & \Rightarrow w_{ii} < \tfrac{h}{|\mathbf{a}_i^T\boldsymbol{\lambda}_3^\star|}, \end{aligned} \tag{C.1}$$

and since $\mathbf{a}_i^T\boldsymbol{\lambda}_3^\star \leq \|\mathbf{a}_i\|_2\|\boldsymbol{\lambda}_3^\star\|_2 = \|\mathbf{a}_i\|_2$, we get

$$w_{ii} < \frac{h}{\|\mathbf{a}_i\|_2} \Rightarrow q_i = 0. \tag{C.2}$$

Appendix D

Proof of Proposition 2

The denominator of (3.16) can be rewritten in terms of the signal subspace $\mathbf{E}_S$ as

$$\|(\mathbf{I} - \mathbf{E}_S\mathbf{E}_S^T)\mathbf{a}_i\|_2^2. \tag{D.1}$$

On assuming that the signal subspace $\mathbf{E}_S$ is spanned by the set of columns that belong to the support of the underlying sparse signal, the denominator in (D.1) can be rewritten as

$$\|\mathbf{E}_N^T\mathbf{a}_i\|_2^2 = \|(\mathbf{I} - \mathbf{A}_s(\mathbf{A}_s^T\mathbf{A}_s)^{-1}\mathbf{A}_s^T)\mathbf{a}_i\|_2^2, \tag{D.2}$$

where $\mathbf{A}_s$ is the matrix with columns that belong to the support of the sparse signal. On assuming low mutual coherence among the columns in $\mathbf{A}_s$, one can approximate $(\mathbf{A}_s^T\mathbf{A}_s)^{-1}$ as the identity matrix, thus getting

$$\|\mathbf{E}_N^T\mathbf{a}_i\|_2^2 = \|(\mathbf{I} - \mathbf{A}_s\mathbf{A}_s^T)\mathbf{a}_i\|_2^2 = t. \tag{D.3}$$

Let us assume that matrix $\mathbf{A}$ has independent random entries, each distributed as $N(0, \frac{1}{m})$ [13]. Then, the diagonal elements of $\mathbf{A}_s\mathbf{A}_s^T$ follow Chi-square distribution with k degrees of freedom, with mean of $\frac{k}{m}$ and variance $\frac{2k}{m^2}$. Also, the off diagonal elements follow Gamma-variance distribution with mean zero, and variance $\frac{k}{m^2}$. The identity matrix inside the parenthesis of (D.3) has no effect on the statistics of the off diagonal elements of $(\mathbf{A}_s\mathbf{A}_s^T)$, and it only affects the mean of the diagonal entries. The mean of the diagonal entires of $(\mathbf{I} - \mathbf{A}_s^T\mathbf{A}_s)$ is $1 - \frac{k}{m^2}$.

Multiplication of $(\mathbf{I} - \mathbf{A}_s^T\mathbf{A}_s)$ with $\mathbf{a}_i$ results in a vector with independent entries, and each entry is the sum of m random variables. Via the Central Limit Theorem, the resulting random variable can be approximated as a Gaussian with zero mean and variance $\frac{k+m^2-mk+k^2}{m^3}$. On invoking the basic result related to the expected length square of a random Gaussian vector [91], completes the proof.

Appendix E

Variational Approximation of $p(\mathbf{x})$

In this Appendix, we show the intermediate steps to obtain the result of (3.33). Starting with (3.32), Substituting x_i^2 with v, and taking the log, the conjugate function $p^*(\lambda)$ can be found using

$$p^*(\lambda) = \max_v \left(\lambda v - \log(C) + (a + .5)[\log(b + \frac{v}{2})]\right). \tag{E.1}$$

Differentiating (E.1) w.r.t. v, and equating to zero, we get

$$u = -2\,b - \frac{a + 0.5}{\lambda}. \tag{E.2}$$

Substituting (E.2) into (E.1), we get

$$p^*(\lambda) = -\log\left(\frac{\sqrt{2}\,b^a\,\Gamma\left(a + \frac{1}{2}\right)}{2\,\sqrt{\pi}\,\Gamma\left(a\right)\,\left(-\frac{a+\frac{1}{2}}{2\,\lambda}\right)^{a+\frac{1}{2}}}\right) - \lambda\,\left(2\,b + \frac{a + \frac{1}{2}}{\lambda}\right). \tag{E.3}$$

Observing that $f(v)$ is monotonically decreasing function, we only need to consider $\lambda \leq 0$ for maximization of (2.11). Using the monotonically increasing transformation $\lambda = \frac{-1}{2g_i}$, we have

$$p(v) = \max_{g_i \geq 0} \log\left(\frac{\sqrt{2}\,b^a\,\Gamma\left(a + \frac{1}{2}\right)}{2\,\sqrt{\pi}\,\Gamma\left(a\right)\,\frac{\left(g_i\left(a+\frac{1}{2}\right)\right)^{a+\frac{1}{2}}}{2}}\right) - \frac{v}{g_i} - \frac{2\,b - g_i\,\left(a + \frac{1}{2}\right)}{g_i}. \tag{E.4}$$

Taking the exponent of both sides of (E.4), substituting x_i^2 with v, we get

$$p(x_i) = \max_{g_i \geq 0} \frac{2^{3\,a-1}\,b^a\,\Gamma\left(a + \frac{1}{2}\right)^2\,\mathrm{e}^{-\frac{w^2 + 2\,b - g - 2\,a\,g}{2\,g}}}{g^{a+\frac{1}{2}}\,\pi\,\Gamma\left(2\,a\right)\,\left(2\,a + 1\right)^{a+\frac{1}{2}}}. \tag{E.5}$$

References

[1] S. C. Wu and A. L. Swindlehurst, "Matching pursuit and source deflation for sparse EEG/MEG dipole moment estimation," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 8, pp. 2280–2288, 2013.

[2] A. Al Hilli, L. Najafizadeh, and A. Petropulu, "A Weighted Approach for Sparse Signal Support Estimation with Application to EEG Source Localization," *IEEE Transactions on Signal Processing*, 2017.

[3] M. Lustig, D. L. Donoho, J. M. Santos, and J. M. Pauly, "Compressed sensing MRI," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 72–82, 2008.

[4] S. Aviyente, "Compressed sensing framework for EEG compression," in *Proc. IEEE/SP 14th Workshop Stat. Signal Process*, 2007, pp. 181–184.

[5] J. P. Haldar, D. Hernando, and Z.-P. Liang, "Compressed-sensing MRI with random encoding," *IEEE Transactions on Medical Imaging*, vol. 30, no. 4, pp. 893–903, 2011.

[6] Y. Yu, A. P. Petropulu, and H. V. Poor, "MIMO radar using compressive sampling," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 1, pp. 146–163, 2010.

[7] A. Al Hilli, L. Najafizadeh, and A. Petropulu, "Sparse target scene reconstruction for SAR using range space rotation," in *IEEE Radar Conference (RadarConf)*, 2016, pp. 1–5.

[8] M. Herman and T. Strohmer, "Compressed sensing radar," in *Proc. IEEE Radar Conference*, 2008, pp. 1–6.

[9] J. H. Ender, "On compressive sensing applied to radar," *Signal Processing*, vol. 90, no. 5, pp. 1402–1414, 2010.

[10] M. T. Alonso, P. López-Dekker, and J. J. Mallorquí, "A novel strategy for radar imaging based on compressive sensing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 12, pp. 4285–4295, 2010.

[11] A. Al Hilli and A. Petropulu, "MIMO radar using sparse sensing: A weighted sparse Bayesian learning (WSBL) approach," in *Signals, Systems, and Computers, 2017 51st Asilomar Conference on.* IEEE, 2017, pp. 80–84.

[12] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM journal on computing*, vol. 24, no. 2, pp. 227–234, 1995.

[13] E. J. Candes and T. Tao, "Decoding by linear programming," *IEEE Transactions on Information Theory*, vol. 51, no. 12, pp. 4203–4215, 2005.

[14] A. Cohen, W. Dahmen, and R. DeVore, "Compressed sensing and best k-term approximation," *Journal of The American Mathematical Society*, vol. 22, no. 1, pp. 211–231, 2009.

[15] D. L. Donoho and M. Elad, "Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ_1 minimization," *Proceedings of the National Academy of Sciences*, vol. 100, no. 5, pp. 2197–2202, 2003.

[16] Y.-B. Zhao, "Rsp-based analysis for sparsest and least-norm solutions to underdetermined linear systems," *IEEE Transactions on Signal Processing*, vol. 61, no. 22, pp. 5777–5788, 2013.

[17] I. F. Gorodnitsky and B. D. Rao, "Sparse signal reconstruction from limited data using FOCUSS: A re-weighted minimum norm algorithm," *IEEE Transactions on Signal Processing*, vol. 45, no. 3, pp. 600–616, 1997.

[18] E. J. Candes, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5-6, pp. 877–905, 2008.

[19] D. P. Wipf and B. D. Rao, "Sparse bayesian learning for basis selection," *IEEE Transactions on Signal Processing*, vol. 52, no. 8, pp. 2153–2164, 2004.

[20] I. F. Gorodnitsky and B. D. Rao, "Sparse signal reconstruction from limited data using FOCUSS: A re-weighted minimum norm algorithm," *IEEE Transactions on Signal Processing*, vol. 45, no. 3, pp. 600–616, 1997.

[21] B. D. Rao, K. Engan, S. F. Cotter, J. Palmer, and K. Kreutz-Delgado, "Subset selection in noise based on diversity measure minimization," *IEEE Transactions on Signal Processing*, vol. 51, no. 3, pp. 760–770, 2003.

[22] P. Xu, Y. Tian, H. Chen, and D. Yao, "Lp norm iterative sparse solution for EEG source localization," *IEEE Transactions on Biomedical Engineering*, vol. 54, no. 3, pp. 400–409, 2007.

[23] S. F. Cotter, R. Adler, R. Rao, and K. Kreutz-Delgado, "Forward sequential algorithms for best basis selection," *IEE Proceedings-Vision, Image and Signal Processing*, vol. 146, no. 5, pp. 235–244, 1999.

[24] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.

[25] J. Adler, B. D. Rao, and E. Kreutz-Delgado, "Comparison of basis selection methods," in *Signals, Systems and Computers, 1996. Conference Record of the Thirtieth Asilomar Conference on*, vol. 1. IEEE, 1996, pp. 252–257.

[26] M. E. Tipping, "Sparse bayesian learning and the relevance vector machine," *The journal of machine learning research*, vol. 1, pp. 211–244, 2001.

[27] J. M. Bernardo and A. F. Smith, "Bayesian theory," 2001.

[28] M. Figueiredo, "Adaptive sparseness using jeffreys prior," in *Advances in neural information processing systems*, 2002, pp. 697–704.

[29] J. Fang, Y. Shen, F. Li, H. Li, and Z. Chen, "Support knowledge-aided sparse bayesian learning for compressed sensing," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on.* IEEE, 2015, pp. 3786–3790.

[30] D. P. Wipf and B. D. Rao, "An empirical bayesian strategy for solving the simultaneous sparse approximation problem," *IEEE Transactions on Signal Processing*, vol. 55, no. 7, pp. 3704–3716, 2007.

[31] F. Santosa and W. W. Symes, "Linear inversion of band-limited reflection seismograms," *SIAM Journal on Scientific and Statistical Computing*, vol. 7, no. 4, pp. 1307–1330, 1986.

[32] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM review*, vol. 43, no. 1, pp. 129–159, 2001.

[33] Y.-B. Zhao, "RSP-Based Analysis for Sparsest and Least-Norm Solutions to Underdetermined Linear Systems," *IEEE Transactions on Signal Processing*, vol. 61, no. 22, pp. 5777–5788, 2013.

[34] D. L. Donoho and M. Elad, "Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ_1 minimization," *Proceedings of the National Academy of Sciences*, vol. 100, no. 5, pp. 2197–2202, 2003.

[35] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33–61, 1998.

[36] E. J. Candès *et al.*, "Compressive sampling," in *Proceedings of The International Congress of Mathematicians*, vol. 3. Madrid, Spain, 2006, pp. 1433–1452.

[37] M. Babaie-Zadeh, B. Mehrdad, and G. B. Giannakis, "Weighted sparse signal decomposition," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on.* IEEE, 2012, pp. 3425–3428.

[38] F. Tadel, S. Baillet, J. C. Mosher, D. Pantazis, and R. M. Leahy, "Brainstorm: a user-friendly application for MEG/EEG analysis," *Computational Intelligence and Neuroscience*, vol. 2011, p. 8, 2011.

[39] C. S. Herrmann, M. H. Munk, and A. K. Engel, "Cognitive functions of gamma-band activity: memory match and utilization," *Trends in Cognitive Sciences*, vol. 8, no. 8, pp. 347–355, 2004.

[40] S. Ponten, A. Daffertshofer, A. Hillebrand, and C. J. Stam, "The relationship between structural and functional connectivity: graph theoretical analysis of an EEG neural mass model," *NeuroImage*, vol. 52, no. 3, pp. 985–994, 2010.

[41] N. Karamzadeh, A. Medvedev, A. Azari, A. Gandjbakhche, and L. Najafizadeh, "Capturing dynamic patterns of task-based functional connectivity with EEG," *NeuroImage*, vol. 66, pp. 311–317, 2013.

[42] B. J. Roach and D. H. Mathalon, "Event-related EEG time-frequency analysis: an overview of measures and an analysis of early gamma band phase locking in schizophrenia," *Schizophrenia Bulletin*, vol. 34, no. 5, pp. 907–926, 2008.

[43] L. Yang, C. Wilke, B. Brinkmann, G. A. Worrell, and B. He, "Dynamic imaging of ictal oscillations using non-invasive high-resolution EEG," *Neuroimage*, vol. 56, no. 4, pp. 1908–1917, 2011.

[44] D. Moretti, D. Paternicò, G. Binetti, O. Zanetti, and G. B. Frisoni, "EEG markers are associated to gray matter changes in thalamus and basal ganglia in subjects with mild cognitive impairment," *NeuroImage*, vol. 60, no. 1, pp. 489–496, 2012.

[45] L. Qin, L. Ding, and B. He, "Motor imagery classification by means of source analysis for brain–computer interface applications," *Journal of Neural Engineering*, vol. 1, no. 3, p. 135, 2004.

[46] B. Blankertz, G. Dornhege, M. Krauledat, K.-R. Müller, V. Kunzmann, F. Losch, and G. Curio, "The Berlin Brain-Computer Interface: EEG-based communication without subject training," *IEEE Transactions on Neural Systems and Rehabilitation Engineering,*, vol. 14, no. 2, pp. 147–152, 2006.

[47] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, and B. Arnaldi, "A review of classification algorithms for EEG-based brain–computer interfaces," *Journal of Neural Engineering*, vol. 4, no. 2, p. R1, 2007.

[48] R. Xu, N. Jiang, C. Lin, N. Mrachacz-Kersting, K. Dremstrup, and D. Farina, "Enhanced low-latency detection of motor intention from EEG for closed-loop brain-computer interface applications," *IEEE Transactions on Biomedical Engineering*, vol. 61, no. 2, pp. 288–296, 2014.

[49] P. L. Nunez, R. Srinivasan, A. F. Westdorp, R. S. Wijesinghe, D. M. Tucker, R. B. Silberstein, and P. J. Cadusch, "EEG coherency: I: statistics, reference electrode, volume conduction, Laplacians, cortical imaging, and interpretation at multiple scales," *Electroencephalography and Clinical Neurophysiology*, vol. 103, no. 5, pp. 499–515, 1997.

[50] G. Pfurtscheller, C. Brunner, A. Schlögl, and F. L. Da Silva, "Mu rhythm (de) synchronization and EEG single-trial classification of different motor imagery tasks," *NeuroImage*, vol. 31, no. 1, pp. 153–159, 2006.

[51] S. Ge, R. Wang, and D. Yu, "Classification of four-class motor imagery employing single-channel electroencephalography," *PLoS ONE*, vol. 9, no. 6, p. e98019, 2014.

[52] F. Shiman, E. López-Larraz, A. Sarasola-Sanz, N. Irastorza-Landa, M. Spueler, N. Birbaumer, and A. Ramos-Murguialday, "Classification of different reaching movements from the same limb using eeg," *Journal of neural engineering*, 2017.

[53] B. J. Edelman, B. Baxter, and B. He, "EEG Source Imaging Enhances the Decoding of Complex Right-Hand Motor Imagery Tasks," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 1, pp. 4–14, 2016.

[54] R. Grech, T. Cassar, J. Muscat, K. P. Camilleri, S. G. Fabri, M. Zervakis, P. Xanthopoulos, V. Sakkalis, and B. Vanrumste, "Review on solving the inverse problem in EEG source analysis," *Journal of Neuroengineering and Rehabilitation*, vol. 5, no. 1, p. 1, 2008.

[55] R. D. Pascual-Marqui, "Review of methods for solving the EEG inverse problem," *International Journal of Bioelectromagnetism*, vol. 1, no. 1, pp. 75–86, 1999.

[56] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, 1986.

[57] J. C. Mosher, P. S. Lewis, and R. M. Leahy, "Multiple dipole modeling and localization from spatio-temporal MEG data," *IEEE Transactions on Biomedical Engineering*, vol. 39, no. 6, pp. 541–557, 1992.

[58] J. Capon, "High-resolution frequency-wavenumber spectrum analysis," *Proceedings of the IEEE*, vol. 57, no. 8, pp. 1408–1418, 1969.

[59] K. Sekihara, S. S. Nagarajan, D. Poeppel, A. Marantz, and Y. Miyashita, "Reconstructing spatio-temporal activities of neural sources using an MEG vector beamformer technique," *IEEE Transactions on Biomedical Engineering*, vol. 48, no. 7, pp. 760–771, 2001.

[60] O. L. Frost, "An algorithm for linearly constrained adaptive array processing," *Proceedings of the IEEE*, vol. 60, no. 8, pp. 926–935, 1972.

[61] B. D. Van Veen, W. Van Drongelen, M. Yuchtman, and A. Suzuki, "Localization of brain electrical activity via linearly constrained minimum variance spatial filtering," *IEEE Transactions on Biomedical Engineering*, vol. 44, no. 9, pp. 867–880, 1997.

[62] A. Rodríguez-Rivera, B. V. Baryshnikov, B. D. Van Veen, and R. T. Wakai, "MEG and EEG source localization in beamspace," *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 3, pp. 430–441, 2006.

[63] K. Matsuura and Y. Okabe, "Selective minimum-norm solution of the biomagnetic inverse problem," *IEEE Transactions on Biomedical Engineering*, vol. 42, no. 6, pp. 608–615, 1995.

[64] ——, "A robust reconstruction of sparse biomagnetic sources," *IEEE Transactions on Biomedical Engineering*, vol. 44, no. 8, pp. 720–726, 1997.

[65] L. Ding and B. He, "Sparse Source Imaging in EEG," in *Joint Meeting of the 6th International Symposium on Noninvasive Functional Source Imaging of the Brain and Heart and the International Conference on Functional Biomedical Imaging*, 2007, pp. 20–23.

[66] K. Uutela, M. Hämäläinen, and E. Somersalo, "Visualization of magnetoencephalographic data using minimum current estimates," *NeuroImage*, vol. 10, no. 2, pp. 173–180, 1999.

[67] S. C. Wu and A. L. Swindlehurst, "Matching pursuit and source deflation for sparse EEG/MEG dipole moment estimation," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 8, pp. 2280–2288, 2013.

[68] H. Hallez, B. Vanrumste, R. Grech, J. Muscat, W. De Clercq, A. Vergult, Y. D'Asseler, K. P. Camilleri, S. G. Fabri, S. Van Huffel *et al.*, "Review on solving the forward problem in EEG source analysis," *Journal of Neuroengineering and Rehabilitation*, vol. 4, no. 1, p. 1, 2007.

[69] "http://imaging.mrc-cbu.cam.ac.uk/imaging/MniTalairach."

[70] B. He, L. Yang, C. Wilke, and H. Yuan, "Electrophysiological imaging of brain activity and connectivitychallenges and opportunities," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 7, pp. 1918–1931, 2011.

[71] M. Fuchs, R. Drenckhahn, H. Wischmann, and M. Wagner, "An improved boundary element method for realistic volume-conductor modeling," *IEEE Transactions on Biomedical Engineering*, vol. 45, no. 8, pp. 980–997, 1998.

[72] A. Delorme and S. Makeig, "EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis," *Journal of Neuroscience Methods*, vol. 134, no. 1, pp. 9–21, 2004.

[73] K. Sekihara, K. E. Hild, and S. S. Nagarajan, "A novel adaptive beamformer for MEG source reconstruction effective when large background brain activities exist," *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 9, pp. 1755–1764, 2006.

[74] G. P. Jacobson and M. B. Fitzgerald, "Auditory evoked gamma band potential in normal subjects," *Journal-American Academy of Audiology*, vol. 8, pp. 44–52, 1997.

[75] J. Schadow, D. Lenz, S. Thaerig, N. A. Busch, I. Fründ, and C. S. Herrmann, "Stimulus intensity affects early sensory processing: sound intensity modulates auditory evoked gamma-band activity in human EEG," *International Journal of Psychophysiology*, vol. 65, no. 2, pp. 152–161, 2007.

[76] K. Whittingstall, G. Stroink, and B. Dick, "Dipole localization accuracy using grand-average EEG data sets," *Clinical Neurophysiology*, vol. 115, no. 9, pp. 2108–2112, 2004.

[77] J. Yao and J. P. Dewald, "Evaluation of different cortical source localization methods using simulated and experimental EEG data," *Neuroimage*, vol. 25, no. 2, pp. 369–382, 2005.

[78] S. D. Mayhew, S. G. Dirckx, R. K. Niazy, G. D. Iannetti, and R. G. Wise, "EEG signatures of auditory activity correlate with simultaneously recorded fMRI responses in humans," *Neuroimage*, vol. 49, no. 1, pp. 849–864, 2010.

[79] L. Jäncke, N. Shah, S. Posse, M. Grosse-Ryuken, and H.-W. Müller-Gärtner, "Intensity coding of auditory stimuli: an fMRI study," *Neuropsychologia*, vol. 36, no. 9, pp. 875–883, 1998.

[80] Y.-T. Zhang, Z.-J. Geng, Q. Zhang, W. Li, and J. Zhang, "Auditory cortical responses evoked by pure tones in healthy and sensorineural hearing loss subjects: functional MRI and magnetoencephalography." *Chinese Medical Journal*, vol. 119, no. 18, pp. 1548–1554, 2006.

[81] E. Fishler, A. Haimovich, R. Blum, D. Chizhik, L. Cimini, and R. Valenzuela, "Mimo radar: An idea whose time has come," in *Radar Conference, 2004. Proceedings of the IEEE*. IEEE, 2004, pp. 71–78.

[82] L. Xu, J. Li, and P. Stoica, "Radar imaging via adaptive mimo techniques," in *Signal Processing Conference, 2006 14th European*. IEEE, 2006, pp. 1–5.

[83] M. Schneider, "Automotive radar–status and trends," in *German microwave conference*, 2005, pp. 144–147.

[84] A. Al Hilli, L. Najafizadeh, and A. Petropulu, "Weighted sparse Bayesian learning (WSBL) for basis selection in linear underdetermined systems," in *Compressed Sensing Theory and its Applications to Radar, Sonar and Remote Sensing (CoSeRa), 2016 4th International Workshop on*. IEEE, 2016, pp. 115–119.

[85] K. Lee, Y. Bresler, and M. Junge, "Subspace methods for joint sparse recovery," *IEEE Transactions on Information Theory*, vol. 58, no. 6, pp. 3613–3641, 2012.

[86] A. Papoulis and S. U. Pillai, *Probability, random variables, and stochastic processes*. Tata McGraw-Hill Education, 2002.

[87] N. Samuel, T. Diskin, and A. Wiesel, "Learning to detect," *arXiv preprint arXiv:1805.07631*, 2018.

[88] M. Carlin, P. Rocca, G. Oliveri, F. Viani, and A. Massa, "Directions-of-arrival estimation through bayesian compressive sensing strategies," *IEEE Transactions on Antennas and Propagation*, vol. 61, no. 7, pp. 3828–3838, 2013.

[89] A. A. Hilli and A. Petropulu, "MIMO radar using sparse sensing: A weighted sparse Bayesian learning (WSBL) approach," in *2017 51st Asilomar Conference on Signals, Systems, and Computers*, Oct 2017, pp. 80–84.

[90] P. Langley, W. Iba, K. Thompson *et al.*, "An analysis of bayesian classifiers," in *Aaai*, vol. 90, 1992, pp. 223–228.

[91] (https://math.stackexchange.com/users/6179/did), "Expected distance between two vectors that belong to two different gaussian distributions," Mathematics Stack Exchange, URL:https://math.stackexchange.com/q/917890 (version: 2014-09-03). [Online]. Available: https://math.stackexchange.com/q/917890