HIERARCHICAL SPARSE MODELING USING SPIKE AND SLAB PRIORS

*Yuanming Suo, Minh Dao, Trac Tran**

Electrical and Computer Engineering
Johns Hopkins University

Umamahesh Srinivas, Vishal Monga

Electrical Engineering
Pennsylvania State University

ABSTRACT

Sparse modeling has demonstrated its superior performances in many applications. Compared to optimization based approaches, Bayesian sparse modeling generally provides a more sparse result with a knowledge of confidence. Using the Spike and Slab priors, we propose the hierarchical sparse models for the scenario of single task and multitask - Hi-BCS and CHi-BCS. We draw the connections of these two methods to their optimization based counterparts and use expectation propagation for inference. The experiment results using synthetic and real data demonstrate that the performance of Hi-BCS and Chi-BCS are comparable or better than their optimization based counterparts.

Index Terms— sparse modeling, hierarchical, spike and slab, compressed sensing

1. INTRODUCTION

In the fields across science and engineering, we're facing increasing problems dealing with high dimensional signals, which are often inherently sparse in some dictionary. The candidate for the dictionary could be fourier, wavelet or some trained dictionaries. Compressed Sensing (CS) [1, 2] leverages this sparsity structure and has gained success in applications such as face recognition, image denoising and video tracking. In particular, Lasso [3] is proposed to minimize the squared loss subject to l_1 constraints on the sparse coefficients. Elastic Net [4] penalizes both the l_1 norm and l_2 norm to encourage grouping. With known hierarchical structure, Group Lasso [5, 6] penalizes the group level sparsity. However, Group Lasso tends to produces results that are dense inside each group. Thus, Hierarchical Lasso (HiLasso) is proposed to regualize both the group sparsity and in-group sparsity. And its multi-task version Collaborative HiLasso (C-HiLasso) takes into account of both the group structure in each task and block struture across multiple tasks [7].

Besides these optimization based approaches, Bayesian inference has also been used for sparse modeling. The benefits of using Bayesian inference are two-folded. First, it provides the full posterior of sparse coefficients rather than a point estimate, which can provide us a knowledge of confidence in the result. Second, the Bayesian framework generally gives more sparse results compared to optimization based approaches [8, 9, 10]. Lasso can be interpreted as a MAP estimate when the sparse coefficients have independent Laplace priors [11]. To perform Bayesian inference in a closed form, Bayesian Compressive Sensing (BCS) [8] adopts a hierarchical Gaussian-Gamma prior instead (which corresponds to the Student-t distribution). BCS has also been extended to the multi-task case (MT-BCS) [12] by sharing the priors among different tasks and gain more effectiveness and robustness. Besides Laplacian and Student-t priors, a particularly well-suited example of sparsity prior is the Spike and Slab prior [13, 14]. Spike and Slab has become a gold standard for sparsity prior for two reasons. First, it's more effective in enforcing sparsity by selectively reducing the magnitude of a subset of the sparse coefficients while both Laplace and Student-t have a single characteristic scale. Second, the desired degree of sparsity is related to the weight for the Spike part.

This paper is divided into the following sections: In section II, we discuss the relation of our work to prior work. In section III, we will further elaborate on the advantages of using Spike and Slab prior and bridge the gap between Bayesian sparse modeling using Spike and Slab and optimization based approaches. Then we show that by modifying the priors of sparse coefficients, we will have the Bayesian counterparts of HiLasso and C-HiLasso. In section IV we compare the performance of Bayesian version of HiLasso and C-HiLasso with the optimization based approaches using synthetic data and real data. We end the paper with a conclusion and future work in section V.

2. RELATION TO PRIOR WORK

Motivated by the benefits of Bayesian sparse modeling and advantages of Spike and Slab priors, we focus on the hierarchical sparse modeling using Spike and Slab priors for both single task and multitask scenarios. The connection between sparse modeling using Spike and Slab prior with Elastic Net is established in [14]. We give a further analysis on the regularization parameters. Also we propose two hierarchical sparse models using Spike and Slab priors and relate them to Hi-

*This work has been partially supported by NSF under Grant CCF-1117545, ARO under Grant 60219-MA, and ONR under Grant N000141210765.

Lasso and C-HiLasso. Prior works using Spike and Slab priors for multi-task learning problems [15, 16] only consider the block sparsity among different tasks while our work consider both the block sparsity across tasks and the group sparsity inside each task.

3. HIERARCHICAL SPARSE MODELING USING SPIKE AND SLAB PRIORS

In the single task scenario, we have the dictionary $\boldsymbol{A}$, the measurement $\boldsymbol{y}$ and the corresponding sparse coefficient $\boldsymbol{x}$, which has a signal model as:

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{n}, \tag{1}$$

where $\boldsymbol{y} \in \mathbb{R}^m, \boldsymbol{x} \in \mathbb{R}^n, \boldsymbol{A} \in \mathbb{R}^{m \times n}$, and $\boldsymbol{n}$ models the Gaussian noise. For Bayesian Sparse Modeling using Spike and Slab priors (with independence assumption), we have:

$$\boldsymbol{y}|\boldsymbol{A}, \boldsymbol{x}, \boldsymbol{\gamma}, \sigma_n^2 \sim \mathcal{N}\left(\boldsymbol{A}\boldsymbol{x}, \sigma_n^2 \boldsymbol{I}\right) \tag{2}$$

$$\boldsymbol{x}|\sigma^2, \boldsymbol{\gamma} \sim \prod_{i=1}^{n} \gamma_i \mathcal{N}(0, \sigma^2) + (1-\gamma_i)\delta_0 \tag{3}$$

$$\boldsymbol{\gamma}|\kappa \sim \prod_{i=1}^{n} \text{Bernoulli}(\kappa). \tag{4}$$

where $\boldsymbol{\gamma}$ is the latent variable indicating the chosen support, σ_n^2 denotes the noise standard deviation, σ^2 represents the spread of the Slab part and κ reflects the sparsity. It is shown in [14] that with fixed σ_n^2, σ^2 and κ, the cost function for MAP estimate reduces to:

$$L(\boldsymbol{x}) = \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \lambda \|\boldsymbol{x}\|_2^2 + \rho \|\boldsymbol{x}\|_0, \tag{5}$$

where $\rho = \sigma_n^2 \log\left(\frac{2\pi\sigma^2(1-\kappa)^2}{\kappa^2}\right)$ and $\lambda = \frac{\sigma_n^2}{\sigma^2}$. This cost function is similar to the cost function of Elastic Net [4], while the latter one has to relax l_0 norm to l_1 norm for optimization purposes. Thus Spike and Slab priors can lead to a more sparse result while having the grouping benefit of Elastic Net.

To bring some intuition, we take one more step to look at the regularization parameters λ and ρ. When the noise power and sparsity of the signal are fixed (fixed σ_n^2 and κ), if we're looking for a solution with more small energy terms (smaller spread of Slab part σ^2) rather than with a few large energy terms, λ would become larger while ρ becomes smaller to encourage the l_2 norm. When the noise power and the spread of Slab part are fixed (fixed σ_n^2 and σ^2), if we're looking for a more sparse solution (smaller κ), ρ will become larger to bring more penalty on the l_0 norm. Finally, when the spread of Slab part and the sparsity are fixed (fixed σ^2 and κ), if noise power is larger (larger σ_n^2), we need to emphasis more on the l_0 norm and l_2 norm to find the desired solution. Note that for optimization based approaches, the regularization parameters are always chosen by cross-validation while the Bayesian approach brings more intuition on the relationship between parameters selection and data characteristics.

3.1. Hierarchical extension - Hi-BCS

When the group structure is known, Group Lasso performs better than Lasso. Motivated by this, we extend the Bayesian model to take into account of Group Sparsity by simply modifying the priors as:

$$\boldsymbol{x}|\sigma^2, \boldsymbol{\gamma} \sim \prod_{i=1}^{G} \prod_{j=1}^{K_i} \gamma_i \mathcal{N}(0, \sigma^2) + (1-\gamma_i)\delta_0 \tag{6}$$

$$\boldsymbol{\gamma}|\kappa \sim \prod_{i=1}^{G} \prod_{j=1}^{K_i} \text{Bernoulli}(\kappa). \tag{7}$$

where G is the total number of groups, K_i denotes the number of dictionary atoms inside each group G_i. Following similar deviations as in [14], it can be shown that, the cost function reduces to:

$$L(\boldsymbol{x}) = \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \lambda \sum_G \|\boldsymbol{x}_{[G]}\|_2^2 + \rho \|\boldsymbol{x}\|_0, \tag{8}$$

where $\sum_G \|\boldsymbol{x}_{[G]}\|_2^2$ is the squared Group Lasso regularizer. Thus we can see that by enforcing the group structure, hierarchical sparse modeling using Spike and Slab priors (Hi-BCS) can enforce both the group sparsity and in-group sparsity like HiLasso. And it's superior to Group Lasso because it enforces the overall sparsity by the l_0 norm on sparse coefficients.

3.2. Collaborative Hierarchical extension - CHi-BCS

Although Spike and Slab priors have been used in multitask sparse modeling [15, 16], only block sparsity (correlation between different tasks) has been considered. In this section, we will provide an extension of collaborative hierarchical sparse modeling using Spike and Slab priors (CHi-BCS). It's known that C-HiLasso [7] suits for the scenarios when multiple tasks share the same group of features while both the number of active groups and the number of atoms in each active group are sparse. And its cost function is as follow,

$$L(\boldsymbol{X}) = \|\boldsymbol{Y} - \boldsymbol{A}\boldsymbol{X}\|_2^2 + \lambda \sum_G \|\boldsymbol{X}_{[G]}\|_F + \rho \sum_{t=1}^{T} \|\boldsymbol{x}_t\|_0, \tag{9}$$

where $\boldsymbol{X}_{[G]}$ is the submatrix formed by all the rows belonging to group G and $\sum_G \|\boldsymbol{X}_{[G]}\|_F$ is the Group Lasso regularizer for the case of multitask. Motivated by this, we modify the model to consider the block and group sparsity, which leads to:

$$\boldsymbol{Y}|\boldsymbol{A}, \boldsymbol{X}, \boldsymbol{\gamma}, \sigma_n^2 \sim \prod_{t=1}^{T} \mathcal{N}\left(\boldsymbol{A}\boldsymbol{x}_t, \sigma_n^2 \boldsymbol{I}\right) \tag{10}$$

$$\boldsymbol{X}|\sigma^2, \boldsymbol{\gamma} \sim \prod_{t=1}^{T} \prod_{i=1}^{G} \prod_{j=1}^{K_i} \gamma_i \mathcal{N}(0, \sigma^2) + (1-\gamma_i)\delta_0 \tag{11}$$

$$\boldsymbol{\Gamma}|\kappa \sim \prod_{t=1}^{T} \prod_{i=1}^{G} \prod_{j=1}^{K_i} \text{Bernoulli}(\kappa). \tag{12}$$

where $\boldsymbol{Y}$ and $\boldsymbol{X}$ are the concatenation of $\boldsymbol{y}_t$ and $\boldsymbol{x}_t$. T is the number of tasks. And note that we enforce different tasks to share same $\boldsymbol{\gamma}_i$ for same group G_i so that they will have the desired block structure in the sparse coefficients matrix $\boldsymbol{X}$. The group sparsity is enforced simulataneously on all the groups across all the tasks. Following similar deviations, we can derive the corresponding cost function for CHi-BCS:

$$L(\boldsymbol{X}) = \|\boldsymbol{Y} - \boldsymbol{A}\boldsymbol{X}\|_2^2 + \lambda \sum_G \|\boldsymbol{X}_{[G]}\|_F^2 + \rho \sum_{t=1}^{T} \|\boldsymbol{x}_t\|_0, \quad (13)$$

and we can see the only difference is the square Group Lasso regularizer. Note that the overall sparsity is guranteed by the l_0 norm on individual task. Thus CHi-BCS can enforce both the group sparsity, in-group sparsity and block sparsity.

3.3. Inference

Bayesian inference is computationally demanding for Spike and Slab priors. Different techniques can be used such as sampling methods or approximation methods. We choose expectation propagation (EP) because of its efficiency and demonstrated success for multi-task learning problems [16]. In this paper, we fix the parameters σ^2 to be 1, σ_n^2 to be the true noise variance and κ to be the true sparsity of the signal. Our experience shows that the results are not sensitive to the choice of the parameters. Sampling methods, hyperpriors or EM could also be used to estimate these parameters, but it's beyond the scope of this paper.

Take Hi-BCS for example, we represent the likelihood function (2), the prior for sparse coefficients (6) and the prior for latent variable (7) as different terms t_1, t_2 and t_3. The joint posterior distribution $\mathcal{P}(\boldsymbol{x}, \boldsymbol{\gamma}, \boldsymbol{y}|\boldsymbol{A})$ can be written as the product of these terms. We approximate the posterior with three terms of exponential family parametric distribution for Hi-BCS:

$$\mathcal{Q} = \prod_{i=1}^{G}\prod_{j=1}^{K_i} \mathcal{N}(x_j|m_j, v_j)\text{Bernoulli}(\gamma_i|p_i) \quad (14)$$

$$\tilde{t}_1 = z_1 \prod_{i=1}^{G}\prod_{j=1}^{K_i} \mathcal{N}(x_j|m_{1j}, v_{1j}) \quad (15)$$

$$\tilde{t}_2 = z_2 \prod_{i=1}^{G}\prod_{j=1}^{K_i} \mathcal{N}(x_j|m_{2j}, v_{2j})\text{Bernoulli}(\gamma_i|p_{2i}) \quad (16)$$

$$\tilde{t}_3 = z_3 \prod_{i=1}^{G}\prod_{j=1}^{K_i} \text{Bernoulli}(\gamma_i|p_{3i}), \quad (17)$$

where m_j, v_j (for $j = 1, ..., \sum_G K_i$) and p_i (for $i = 1, ..., G$) are the free parameters to infer and will be our estimate of the mean, variance for sparse coefficients $\boldsymbol{x}$ and mean for latent variables $\boldsymbol{\gamma}$. And m_{1j}, v_{1j}, m_{2j}, v_{2j} (for $j = 1, ..., \sum_G K_i$) and p_{2i} and p_{3i} (for $i = 1, ..., G$) are the parameters to be updated in each EP update. And z_1, z_2 and z_3 are normalization parameters. Note that m_j, v_j is specific for each x_j while p_i is same for the whole group to favor the group sparsity. The complete EP algorithm involves the following steps:

1. Initialize all $\tilde{t}_l$ terms and $\mathcal{Q}$ to be non-informative.
2. Repeat until all the $\tilde{t}_l$ terms converges:

a) To refine each $\tilde{t}_l$ term, first find $\mathcal{Q}^l$ by dividing $\mathcal{Q}$ with $\tilde{t}_l$.
b) Minimize $D_{KL}(t_m\mathcal{Q}^l||\tilde{t}_m\mathcal{Q}^l)$ to modify m_{lj}, v_{lj} and p_{li}
c) Find $\mathcal{Q}$ as the product of the new $\tilde{t}_m$ and $\mathcal{Q}^l$ to update m_j, v_j and p_j

Interested readers can refer to [17] for detailed procedures for exponential family distributions. The EP algorithm for CHi-BCS only modifies the approximation to multitask scenario and will not be provided here due to the page limits.

4. EXPERIMENTAL RESULTS

In this section we compare the performance of the proposed hierarchical sparse modeling Hi-BCS and CHi-BCS with its optimization based counterparts. The regularization parameters for the optimization based approaches are chosen by cross-validation. First, we compare the performance of Hi-BCS to Group Lasso [5] and BCS [8]. We replicate the same 1D signal synthetic data example used in [8]. A signal $\boldsymbol{x}$ of length 512 contains 20 spikes being randomly distributed in 15 groups. Each spike has value 1 or -1. The 100×512 dictionary $\boldsymbol{A}$ has i.i.d Gaussian entries from $\mathcal{N}(0, 1)$ and unit l_2 norm for each row. Zero mean gaussian noise with std of 0.005 is added to $\boldsymbol{y}$. The group structure is given to Group Lasso and Hi-BCS as totally 64 consecutive groups with 8 atoms each group. Fig. 1(b), (c) and (d) demonstrates the

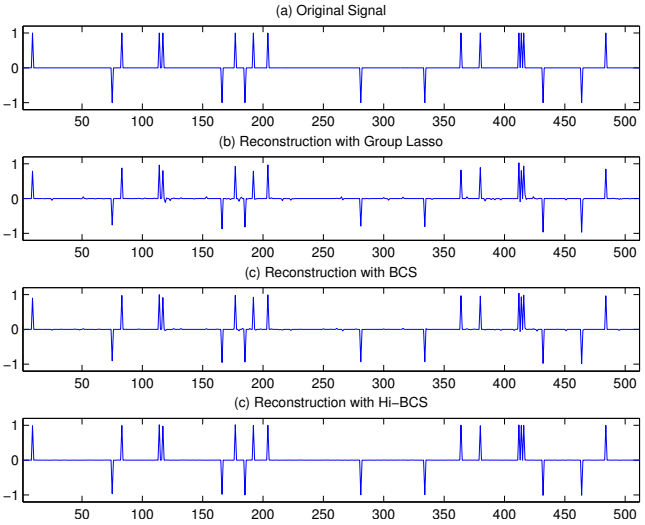

Fig. 1. Reconstruction of uniform spikes for $n = 512$, $m = 100$ and sparsity = 20. (a) Original signal; (b) Reconstruction with BP, $||\boldsymbol{x}_{GroupLasso} - \boldsymbol{x}||_2/||\boldsymbol{x}||_2 = 0.1663$; (c) Reconstruction with BCS, $||\boldsymbol{x}_{BCS} - \boldsymbol{x}||_2/||\boldsymbol{x}||_2 = 0.0669$; (d) Reconstruction with Hi-BCS, $||\boldsymbol{x}_{Hi-BCS} - \boldsymbol{x}||_2/||\boldsymbol{x}||_2 = 0.0133$.

Table 1. Simulated Signal Results. Every 4×4 cell contains MSE ($\times 10^3$) for MT-BCS, C-HiLasso and the proposed CHi-BCS. In the first case (top left), we vary noise σ while keeping $q = 8$, $s = 8$ and $T = 50$. In the second case (top right), we vary group number q while keeping $\sigma = 0$, $s = 8$ and $T = 50$. In the third case (bottom left), we vary in-group sparsity s while keeping $q = 8$, $\sigma = 0$ and $T = 50$. In the last case (bottom right), we vary task number T while keeping $q = 8$, $s = 8$ and $\sigma = 0$. Bold indicates the best results.

noise std σ	MT-BCS	C-HiLasso	CHi-BCS	group number q	MT-BCS	C-HiLasso	CHi-BCS
0.1	28.15	24.82	**23.71**	4	34.94	25.89	**23.91**
0.2	46.16	**29.41**	30.82	8	20.29	21.21	**18.07**
0.4	97.28	**31.25**	50.47	12	**13.08**	16.48	13.93
in-group sparsity s	MT-BCS	C-HiLasso	CHi-BCS	task number T	MT-BCS	C-HiLasso	CHi-BCS
4	8.24	8.18	**1.72**	10	31.71	22.49	**16.99**
8	20.29	21.21	**18.07**	20	24.70	22.25	**17.58**
12	**27.50**	34.59	32.29	50	20.29	21.21	**18.07**

reconstruction results with Group Lasso, BCS and Hi-BCS. All approaches can sucessfully recover the correct support of the signal but Hi-BCS recover with smaller error. It's because of the superior selectivity of Spike and Slab priors and the enforcing both group sparsity and in-group sparsity.

For the multitask scenario, we first compare CHi-BCS with MT-BCS [12] and C-HiLasso [7] using synthetic data. We use SPAMS package for C-HiLasso. As in [7], we create q sub-dictionaries, each with 64 atoms of dimension 64 with i.i.d. Gaussian entries from $\mathcal{N}(0, 1)$ and unit l_2 norm. The dictionary $\boldsymbol{A}$ is built by concatenating the q sub-dictionaries together. We randomly choose the same 2 groups to be active for all the tasks and for each group only s atoms in each group will be active (value 1 for sparse coefficients). Thus each task $\boldsymbol{y}_t$ will be a mixture of $2s$ atoms. And in total we generate T such tasks and add Gaussian noise of standard deviation σ. The true sparse coefficients matrix $\boldsymbol{X}$ has both block and group structure with in-group sparsity. Table I summarizes the mean-square error (MSE) of the recoverd sparse coefficient matrix for different σ, q, s and T. We can see that CHi-BCS performs comparably with C-HiLasso and generally better than MT-BCS. However, when the in-group sparsity s becomes larger, MT-BCS has a slightly better performance than CHi-BCS and C-HiLasso. This is because both CHi-BCS and C-HiLasso favors block sparsity while promoting in-group sparsity.

We also compare the performance of these three methods on real data using the USPS digits dataset. The signals are vectors containing the intensities of 16×16 images of digits 0 to 9 (m = 256). For each digit, we use 150 out of 1100 images to build the test dataset and the other 950 images to train a sub-dictionary. We use K-SVD [18] for dictionary training and the sub-dictionary for each digit has the dimension 256×100. Thus the whole dictionary $\boldsymbol{A}$ has dimension 256×1000. We randomly active 1 atom foreach of the k active digits and generate 10 tasks out of it. Thus each task is the mixture of k randomly chosen different digits and zero mean Gaussian noise with std σ is added. Table II summarizes the MSE of the recoverd coefficients matrix. CHi-BCS works comparably as C-HiLasso and much better than MT-BCS in all scenarios.

Table 2. USPS digits datset Results. Every 4×4 cell contains MSE ($\times 10^3$) for MT-BCS, C-HiLasso and proposed CHi-BCS. In the first case (top), we vary noise σ for $k = 1$. In the second case (bottom), vary noise σ for $k = 2$. Bold indicates the best results.

1 digits	MT-BCS	C-HiLasso	CHi-BCS
$\sigma = 0.1$	1.35	**1.12**	1.25
$\sigma = 0.2$	6.64	1.34	**1.17**
$\sigma = 0.4$	8.66	**0.97**	1.49
2 digits	MT-BCS	C-HiLasso	CHi-BCS
$\sigma = 0.1$	2.98	**2.09**	3.67
$\sigma = 0.2$	8.31	**2.08**	2.27
$\sigma = 0.4$	8.18	**2.12**	2.60

5. CONCLUSION AND DISCUSSION

In this paper, we demonstrate the benefits of using Spike and Slab priors for hierarchical sparse modeling and proposed two extensions for single and multitask scenarios - Hi-BCS and CHi-BCS. The Hi-BCS enforces the group sparsity and in-group sparsity simultaneously. The CHi-BCS can enforce the block and group sparsity while favoring in-group sparsity just like its optimization based counterpart C-HiLasso. These two extensions are shown to have comparable or better performance with its optimization based counterparts using synthetic and real datasets. The result shows once again that enforcing structure prior into sparse modeling can lead to a better results. Note that recent works [19, 20] have shown how to add more sophisticated structure (dirty model or a undirected graph) into the sparse modeling. Thus our future work will focus on extending our approach into more sophisticated structures for different applcations.

6. REFERENCES

[1] E.J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *Information Theory, IEEE Transactions on*, vol. 52, no. 2, pp. 489–509, 2006.

[2] D.L. Donoho, "Compressed sensing," *Information Theory, IEEE Transactions on*, vol. 52, no. 4, pp. 1289–1306, 2006.

[3] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.

[4] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 67, no. 2, pp. 301–320, 2005.

[5] M. Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 68, no. 1, pp. 49–67, 2005.

[6] L. Jacob, G. Obozinski, and J.P. Vert, "Group lasso with overlap and graph lasso," in *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM, 2009, pp. 433–440.

[7] P. Sprechmann, I. Ramirez, G. Sapiro, and Y.C. Eldar, "C-hilasso: A collaborative hierarchical sparse modeling framework," *Signal Processing, IEEE Transactions on*, vol. 59, no. 9, pp. 4183–4198, 2011.

[8] S. Ji, Y. Xue, and L. Carin, "Bayesian compressive sensing," *Signal Processing, IEEE Transactions on*, vol. 56, no. 6, pp. 2346–2356, 2008.

[9] S. Raman, T.J. Fuchs, P.J. Wild, E. Dahl, and V. Roth, "The bayesian group-lasso for analyzing contingency tables," in *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM, 2009, pp. 881–888.

[10] Q. Li and N. Lin, "The bayesian elastic net," *Bayesian Analysis*, vol. 5, no. 1, pp. 151–170, 2010.

[11] T. Park and G. Casella, "The bayesian lasso," *Journal of the American Statistical Association*, vol. 103, no. 482, pp. 681–686, 2008.

[12] S. Ji, D. Dunson, and L. Carin, "Multitask compressive sensing," *Signal Processing, IEEE Transactions on*, vol. 57, no. 1, pp. 92–106, 2009.

[13] H. Ishwaran and J.S. Rao, "Spike and slab variable selection: frequentist and bayesian strategies," *The Annals of Statistics*, vol. 33, no. 2, pp. 730–773, 2005.

[14] T.J. Yen, "A majorization–minimization approach to variable selection using spike and slab priors," *The Annals of Statistics*, vol. 39, no. 3, pp. 1748–1775, 2011.

[15] M. Titsias and M. Lázaro-Gredilla, "Spike and slab variational inference for multi-task and multiple kernel learning," 2011.

[16] D. Hernández-Lobato, J. Hernández-Lobato, T. Helleputte, and P. Dupont, "Expectation propagation for bayesian multi-task feature selection," *Machine Learning and Knowledge Discovery in Databases*, pp. 522–537, 2010.

[17] T.P. Minka, "Expectation propagation for approximate bayesian inference," in *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 2001, pp. 362–369.

[18] M. Aharon, M. Elad, and A. Bruckstein, "K-svd: An algorithm for designing overcomplete dictionaries for sparse representation," *Signal Processing, IEEE Transactions on*, vol. 54, no. 11, pp. 4311–4322, 2006.

[19] A. Jalali, P. Ravikumar, S. Sanghavi, and C. Ruan, "A dirty model for multi-task learning," *Advances in Neural Information Processing Systems*, vol. 23, pp. 964–972, 2010.

[20] J. Miguel Hernández-Lobato, D. Hernández-Lobato, and A. Suárez, "Network-based sparse bayesian classification," *Pattern Recognition*, vol. 44, no. 4, pp. 886–900, 2011.