SAVE - Space Alternating Variational Estimation for Sparse Bayesian Learning

Christo Kurisummoottil Thomas, Dirk Slock

Communication Systems Department, EURECOM

DSW 2018, EPFL, Switzerland

Outline

- Motivation
- State of the Art
- Space Alternating Variational Estimation (SAVE)
- Relation between AMP and SAVE
- Simulation Results

Motivation

- A compressed sensing problem can be formulated as

$$\boldsymbol{y} = \mathbf{A}\boldsymbol{x} + \boldsymbol{w}, \qquad (1)$$

  where $\boldsymbol{y}$ are the observations or data, $\mathbf{A}$ is the $N \times M$ over-complete dictionary matrix which is known and with $N < M$, $\boldsymbol{x}$ is the M-dimensional sparse signal and $\boldsymbol{w}$ is the additive noise. $\boldsymbol{x}$ contains only K non-zero entries, with $K << M$. $\boldsymbol{w}$ is assumed to be a white Gaussian noise, $\boldsymbol{w} \sim \mathcal{N}(0, \gamma^{-1}\boldsymbol{I})$.
- l_0 minimization problem which is an NP-complete problem
- Basis Pursuit [ChenDonoho:SIAM98]: l_1 minimization (convex relaxation of l_0), exact recovery under certain conditions on the over-complete dictionary
- Orthogonal Matching Pursuit (OMP) [TroppGilbert:TIT07], a greedy approach, faster than l_0 and l_1

State of the Art

- The Sparse Bayesian Learning algorithm (SBL) was first introduced by [Tipping:JMLR01] and then proposed for the first time for sparse signal recovery by [WipfRao:TSP04].
- All the above approaches require matrix inversions, $O(M^3)$ complexity
- [TippingFaul:IWAIS03]: Fast Marginalized ML by alternating maximization
- [ShutinBuchgraberKulkarniPoor:T-SP11] Fast SBL by updating alternatingly and using matrix inversion lemmas.
- Both previous approaches allow for a greedy initialization (OMP-like) which improves convergence speed and initialization issues.
- [ShoukairiRao:TSP18] use of Approximate Message Passing (AMP) to approximate matrix inversions in SBL
- [DuanYangFangLi:SPL17] inverse-free SBL via Taylor series expansion

Our Contributions

- We propose a novel Space Alternating Variational Estimation based SBL technique called SAVE.
- We also propose an AMP-style approximation of SAVE, which reveals links to AMP algorithms.
- Numerical results suggest that our proposed solution has a faster convergence rate (and hence lower complexity) than (even) the existing fast SBL and performs better than the existing fast SBL algorithms in terms of reconstruction error in the presence of noise.

Sparse Bayesian Learning (SBL)

- Bayesian CS: 2-layer hierarchical prior for $\boldsymbol{x}$ as in [Tipping:JMLR01], inducing sparsity for $\boldsymbol{x}$.

$$p(\boldsymbol{x}|\boldsymbol{\alpha}) = \prod_{i=1}^{M} p(x_i|\alpha_i) = \prod_{i=1}^{M} \mathcal{N}(x_i; 0, \alpha_i^{-1}).$$

Further a Gamma prior is considered for the precisions α_i,

$$p(\boldsymbol{\alpha}) = \prod_{i=1}^{M} p(\alpha_i/a, b) = \prod_{i=1}^{M} \Gamma^{-1}(a) b^a \alpha_i^{a-1} e^{-b\alpha_i}.$$

- The inverse of the white noise variance γ is also assumed to have a Gamma prior,

$$p(\gamma) = \Gamma^{-1}(c) d^c \gamma^{c-1} e^{-d\gamma}$$

- Marginalizing $\boldsymbol{\alpha}$ leads to student-t distribution for $\boldsymbol{x}$

Variational Bayesian Inference

- The computation of the posterior distribution of the parameters is usually intractable. As in SAGE, SAVE is simply VB with partitioning of the unknowns at the scalar level. Hence the posterior gets approximated as

$$q(\boldsymbol{x}, \boldsymbol{\alpha}, \gamma) = q_{\gamma}(\gamma) \prod_{i=1}^{M} q_{x_i}(x_i) \prod_{i=1}^{M} q_{\alpha_i}(\alpha_i) \tag{2}$$

- Variational Bayes compute the factors q by minimizing the Kullback-Leibler distance between the true posterior distribution $p(\boldsymbol{x}, \boldsymbol{\alpha}, \gamma / \boldsymbol{y})$ and the $q(\boldsymbol{x}, \boldsymbol{\alpha}, \gamma)$. From [Beal:PhD03],

$$KLD_{VB} = KL\left(p(\boldsymbol{x}, \boldsymbol{\alpha}, \gamma / \boldsymbol{y}) || q(\boldsymbol{x}, \boldsymbol{\alpha}, \gamma)\right) \tag{3}$$

Space Alternating Variational Estimation (SAVE)

- The KL divergence minimization is equivalent to maximizing the evidence lower bound (ELBO), $L(q)$ [Tzikas:SPMag08]. To elaborate on this, we can write the marginal probability of the observed data as,

$$\ln p(\boldsymbol{y}) = L(q) + KLD_{VB}, \text{ where,}$$

$$L(q) = \int q(\boldsymbol{\theta}) \ln \frac{p(\boldsymbol{y},\boldsymbol{\theta})}{q(\boldsymbol{\theta})} d\boldsymbol{\theta}, \quad KLD_{VB} = -\int q(\boldsymbol{\theta}) \ln \frac{p(\boldsymbol{\theta}/\boldsymbol{y})}{q(\boldsymbol{\theta})} d\boldsymbol{\theta}. \tag{4}$$

- Let $\boldsymbol{\theta} = \{\boldsymbol{x}, \boldsymbol{\alpha}, \gamma\}$. We get for the element-wise VB recursions

$$\ln(q_i(\theta_i)) = \ < \ln p(\boldsymbol{y},\boldsymbol{\theta}) >_{\bar{i}} + c_i \tag{5}$$

where $<>_{\bar{i}}$ represents the expectation operator with the distributions $q_k()$ for all $k \neq i$. KLD convex in the $q_i(.)$.

SAVE Continued ...

- From $p(\boldsymbol{y}, \boldsymbol{\theta}) = p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\alpha}, \gamma)p(\boldsymbol{x}/\boldsymbol{\alpha})p(\boldsymbol{\alpha})p(\gamma)$ we get

$$\ln p(\boldsymbol{y}, \boldsymbol{\theta}) = \frac{N}{2} \ln \gamma - \frac{\gamma}{2} ||\boldsymbol{y} - \mathbf{A}\boldsymbol{x}||^2 + \sum_{i=1}^{M} \left(\frac{1}{2} \ln \alpha_i - \frac{\alpha_i}{2} x_i^2 \right) + \sum_{i=1}^{M} ((a-1) \ln \alpha_i + a \ln b - b\alpha_i) + (c-1) \ln \gamma + c \ln d - d\gamma + \text{constants},$$

- On the other hand $q(\boldsymbol{\theta}|\boldsymbol{y}) = \prod_{i=1}^{M} q_{\boldsymbol{x}_i}(x_i) \prod_{i=1}^{M} q_{\alpha_i}(\alpha_i)\; q_\gamma(\gamma)$.

SAVE Continued ...

- **Update of $q_{x_i}(x_i)$:**

$$
\begin{aligned}
&\ln q_{x_i}(x_i) = \\
&-\frac{<\gamma>}{2}\Big\{ < ||\boldsymbol{y} - \mathbf{A}_{\bar{i}}\boldsymbol{x}_{\bar{i}}||^2 > -(\boldsymbol{y} - \mathbf{A}_{\bar{i}} < \boldsymbol{x}_{\bar{i}} >)^T \mathbf{A}_i x_i - \\
&x_i \mathbf{A}_i^T(\boldsymbol{y} - \mathbf{A}_{\bar{i}} < \boldsymbol{x}_{\bar{i}} >) + ||\mathbf{A}_i||^2 x_i^2 \Big\} - \frac{<\alpha_i>}{2} x_i^2 + c_{x_i} \\
&= -\frac{1}{2\sigma_i^2}(x_i - \mu_i)^2 + c'_{x_i}.
\end{aligned}
$$

So $q_{x_i}(x_i)$ is Gaussian. Let $\mathbf{A}\boldsymbol{x} = \mathbf{A}_i x_i + \mathbf{A}_{\bar{i}}\boldsymbol{x}_{\bar{i}}$. Then

$$
\sigma_i^2 = \frac{1}{<\gamma>||\mathbf{A}_i||^2 + <\alpha_i>},
$$

$$
\mu_i = <\gamma> \sigma_i^2 A_i^T(\boldsymbol{y} - \mathbf{A}_{\bar{i}} < \boldsymbol{x}_{\bar{i}} >).
$$

- **Update of $q_{\alpha_i}(\alpha_i)$:**

$$\ln q_{\alpha_i}(\alpha_i) = (a - 1 + \tfrac{1}{2}) \ln \alpha_i - \alpha_i \left(\frac{<x_i^2>}{2} + b \right) + c_{\alpha_i},$$
$$\Rightarrow q_{\alpha_i}(\alpha_i) \propto \alpha_i^{1/2} e^{-\alpha_i \left(\frac{<x_i^2>}{2} + b \right)}.$$

The mean of this Gamma distribution is given by

$$< \alpha_i > = \frac{3/2}{\left(\frac{<x_i^2>}{2} + b \right)}, \text{ where } < x_i^2 > = \mu_i^2 + \sigma_i^2.$$

- **Update of $q_\gamma(\gamma)$:**

$$\ln q_\gamma(\gamma) = \frac{N}{2} \ln \gamma - \gamma \left(\frac{<||\mathbf{y} - \mathbf{A}\mathbf{x}||^2>}{2} + d \right) + c_\gamma,$$
$$\Rightarrow q_\gamma(\gamma) \propto \gamma^{N/2} e^{-\gamma \left(\frac{<||\mathbf{y} - A\mathbf{x}||^2>}{2} + d \right)}.$$

The mean of the Gamma distribution for γ is given by

$$< \gamma > = \frac{N/2 + 1}{\left(\frac{<||\mathbf{y} - \mathbf{A}\mathbf{x}||^2>}{2} + d \right)},$$

Computational Complexity

- No matrix inversions
- Update of all the variables, $\boldsymbol{x}, \alpha, \boldsymbol{y}$, requires simple addition and multiplication operations
- $\boldsymbol{y}^T\mathbf{A}$, $\mathbf{A}^T\mathbf{A}$ and $||\boldsymbol{y}||^2$ can be precomputed, so only need to be computed once
- The computational complexity per iteration is of the same order as that of AMP, or Fast MML, or Fast VB

SAVE SBL Algorithm

SAVE SBL Algorithm:

Given: $\boldsymbol{y}, \mathbf{A}, M, N$.
Initialization: b, d are taken to be very low, on the order of $10^{-10}, \boldsymbol{x}^0 = \mathbf{0}, <\alpha_i^0> = \frac{3}{\sigma_i^{2,0}}, <\gamma^0> = \frac{N/2+1}{\left(\frac{<||\boldsymbol{y}||^2>}{2}\right)}$.

Iteration $(t+1)$

- Update $\sigma_i^{2,t+1} = \frac{1}{<\gamma^t>||\mathbf{A}_i||^2 + <\alpha_i^t>}$,
  Point estimate of x_i:
  $x_i^{t+1} = \mu_i = <\gamma^t> \sigma_i^2 A_i^T \left(\boldsymbol{y} - \mathbf{A}_{\bar{i}} <\boldsymbol{x}_{\bar{i}}^t>\right)$
- Compute $<x_i^{2,t+1}> = (x_i^2)^{t+1} + \sigma_i^{2,t+1}$ and update $<\alpha_i^{t+1}> = \frac{3/2}{\left(\frac{<x_i^{2,t+1}>}{2} + b\right)}$,
- Update the noise variance, $<\gamma^{t+1}> = \frac{N/2+1}{\left(\frac{<||\boldsymbol{y} - \mathbf{A}\boldsymbol{x}^t||^2>}{2} + d\right)}$
- Continue till convergence of the algorithm

Relation between AMP and SAVE

- [DonohoMaleki:PNAS09]: first order Approximate Message Passing (AMP) algorithm (from loopy BP) for reconstructing $\boldsymbol{x}$. Starting with an initial guess as $\boldsymbol{x}^{(0)} = \mathbf{0}$,

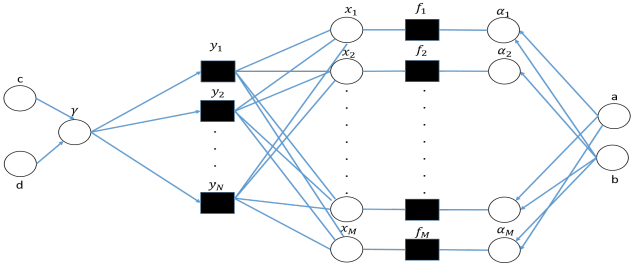
AMP:

In the large system limit, $N, M \to \infty$ with a fixed ratio for $\beta = \frac{N}{M}$ and a possibly non-linear function η_t,

$$\boldsymbol{x}^{t+1} = \eta_t(\mathbf{r}^t),\ \mathbf{r}^t = \mathbf{A}^T\mathbf{z}^t + \boldsymbol{x}^t$$

$$z^t = \boldsymbol{y} - \mathbf{A}\boldsymbol{x}^t + \underbrace{\frac{1}{\beta}\mathbf{z}^{t-1} < \eta'_{t-1}(\mathbf{A}^T\mathbf{z}^{t-1} + \boldsymbol{x}^{t-1}) >}_{Onsager\ \ term}$$

- AMP has been generalized to G-AMP, in which the SVD of $\mathbf{A}$: $\mathbf{A} = \boldsymbol{U}\Sigma\boldsymbol{V}^T$ where $\boldsymbol{U}, \Sigma$ are arbitrary (deterministic) but $\boldsymbol{V}$ is still uniformly unitary (ie Haar distributed)
- For large class of random matrices $\mathbf{A}$, the behaviour of G-AMP can be accurately tracked using state "evolution"



Factor Graph

- $a \in \mathcal{A}$, where $A = \{1, 2....N\}$ represents the indices of the variable nodes y_a and $i \in \mathcal{B}$, where $B = \{1, 2....M\}$ represents the indices of the factor nodes x_i. In the factor graph, factor node f_i represents the computation of the prior distribution of x_i.
- The message for $\boldsymbol{x}$ are Gaussian or for the hyper parameters are Gamma, hence only the means and possibly the variances need to be propagated.

AMP-SAVE Algorithm

- Using first order Taylor series approximations and law of large numbers similar to [DonohoMaleki:PNAS09], we arrive at AMP-SAVE

AMP SAVE Algorithm:

Definitions: $\beta \equiv \frac{N}{M}$, $\mathbf{r}^t \equiv \mathbf{A}^T \mathbf{z}^t + \boldsymbol{x}^t$.

$\mathcal{F}$ operates elementwise, $\mathcal{F}_i(r_i^t) = \frac{\gamma^t}{\alpha_i^t + ||\mathbf{A}_i||^2 \gamma^t} r_i^t$.

Update Equations:

$\boldsymbol{x}^{t+1} = \mathcal{F}(\mathbf{r}^t)$.

$$\mathbf{z}^{t+1} = \boldsymbol{y} - \mathbf{A}\boldsymbol{x}^{t+1} + \underbrace{(1/\beta)\mathbf{z}^t \frac{1}{M} \sum_{j=1}^{M} \frac{\gamma^t}{||\mathbf{A}_i||^2 \gamma^t + \alpha_i^t}}_{Onsager \quad term}.$$

Hyper parameter tuning:

$$\sigma_i^{2,t+1} = \frac{1}{\alpha_i^t + ||\mathbf{A}_i||^2 \gamma^t}, \; \alpha_i^{t+1} = \frac{a + \frac{1}{2}}{\frac{(x_i^{t+1})^2 + \sigma_i^{2,t+1}}{2} + b}, \forall i$$

$$\gamma^{t+1} = \frac{c + \frac{N}{2}}{\left(\frac{||\boldsymbol{y} - \mathbf{A}\boldsymbol{x}^{t+1}||^2 + \mathrm{tr}(\mathbf{A}^T \mathbf{A} \boldsymbol{\Sigma}^{t+1})}{2} + d\right)}.$$

State Evolution

- AMP based algorithms decouple the system of equations into parallel AWGN channels with equal noise variance.
- The quantity $r_i^{t+1} = x_i^t + \mathbf{A}_i^T \mathbf{z}^t$ can be expressed equivalently as $x_i + n_i^t$, where $n_i^t \sim \mathcal{N}(0, \tau_t^2)$ and τ_t^2 is the decoupled noise variance.

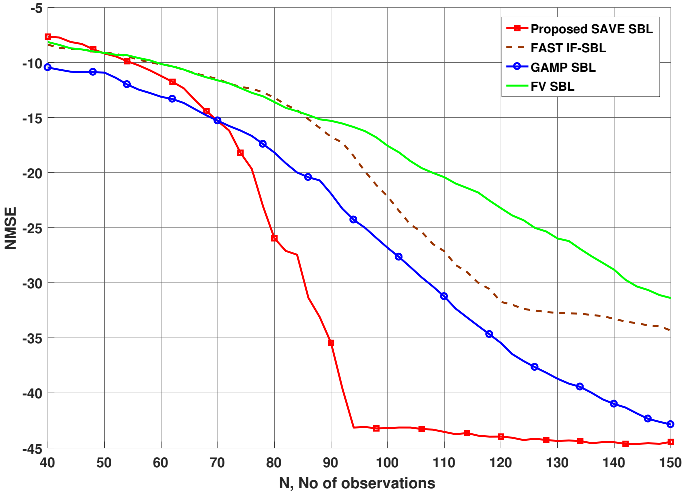
AMP-SAVE State Evolution:

Considering the large system limit and a Lipschitz continuous function $\mathcal{F}$, the decoupled noise variance τ_t^2 and γ^t is given by the following SE recursion, $\tau_{t+1}^2 = \frac{1}{\gamma^{t+1}} + \frac{1}{\beta}\left(\xi^t + \zeta^t \tau_t^2\right)$,
$\frac{1}{\gamma^{t+1}} = \frac{1}{N} ||\boldsymbol{y}||^2 + \frac{1}{\beta}\left(\psi^t + \tau_t^2 \zeta^t\right)$, $\xi^t = \mathrm{E}\left(\frac{\alpha_i^t}{(\gamma^t + \alpha_i^t)^2}\right)$,
$\zeta^t = \mathrm{E}\left(\frac{(\gamma^t)^2}{(\gamma^t + \alpha_i^t)^2}\right), \psi^t = \mathrm{E}\left(\frac{(\gamma^t)^2}{\alpha_i^t(\gamma^t + \alpha_i^t)^2}\right)$.

Simulation Results

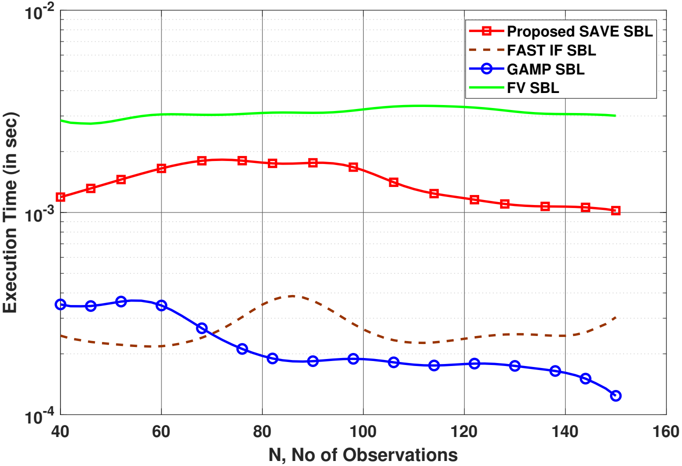
- We compare our algorithm with:
  - The Fast Inverse-Free SBL (Fast IF SBL) in [DuanYangFangLi:SPL17]
  - The G-AMP based SBL in [ShoukairiRao:TSP18]
  - The fast version of SBL (FV SBL) in [Shutin:TSP11]
- Simulations with $M = 200$ and $K = 30$
- All the elements of $\mathbf{A}$ and $\boldsymbol{x}$ are generated i.i.d from a normal distribution, $\mathcal{N}(0, 1)$
- The SNR is fixed to be $20\,dB$ in the simulation

NMSE Results



NMSE vs the number of observations.

Computational Complexity Results



Execution time vs the number of observations.

Conclusion

- SAVE: fast SBL algorithm, which uses the variational inference techniques to approximate the posteriors of the data and parameters.
- SAVE helps to circumvent the matrix inversion operation required in conventional SBL using EM algorithm.
- Proposed algorithm has a faster convergence rate and better performance in terms of NMSE than even the state of the art fast SBL solutions.
- Possible extensions:
  - the case in which $\mathbf{A}$ is parametric in an unknown $\boldsymbol{\theta}$: potential application as wireless channel estimation
  - SBL in the context of multiple measurement vectors case as in [ZhangRao:JSTSP2011], with temporal correlation