Scale Mixture Modeling of Priors for Sparse Signal Recovery

Bhaskar D Rao¹
University of California, San Diego

¹Thanks to David Wipf, Jason Palmer, Zhilin Zhang and Ritwik Giri
Outline

Sparse Signal Recovery (SSR) Problem and some Extensions

Scale Mixture Priors
  - Gaussian Scale Mixture (GSM)
  - Laplacian Scale Mixture (LSM)
  - Power Exponential Scale Mixture (PESM)

Bayesian Methods
  - MAP estimation (Type I)
  - Hierarchical Bayes (Type II)

Experimental Results

Summary

Bhaskar D Rao  University of California, San Diego
Outline

- Sparse Signal Recovery (SSR) Problem and some Extensions
Sparse Signal Recovery (SSR) Problem and some Extensions

Scale Mixture Priors

- Gaussian Scale Mixture (GSM)
- Laplacian Scale Mixture (LSM)
- Power Exponential Scale Mixture (PESM)
Outline

- Sparse Signal Recovery (SSR) Problem and some Extensions
- Scale Mixture Priors
  - Gaussian Scale Mixture (GSM)
  - Laplacian Scale Mixture (LSM)
  - Power Exponential Scale Mixture (PESM)
- Bayesian Methods
  - MAP estimation (Type I)
  - Hierarchical Bayes (Type II)
Outline

- Sparse Signal Recovery (SSR) Problem and some Extensions
- Scale Mixture Priors
  - Gaussian Scale Mixture (GSM)
  - Laplacian Scale Mixture (LSM)
  - Power Exponential Scale Mixture (PESM)
- Bayesian Methods
  - MAP estimation (Type I)
  - Hierarchical Bayes (Type II)
- Experimental Results
Outline

- Sparse Signal Recovery (SSR) Problem and some Extensions
- Scale Mixture Priors
  - Gaussian Scale Mixture (GSM)
  - Laplacian Scale Mixture (LSM)
  - Power Exponential Scale Mixture (PESM)
- Bayesian Methods
  - MAP estimation (Type I)
  - Hierarchical Bayes (Type II)
- Experimental Results
- Summary
Problem Description: Sparse Signal Recovery (SSR)

- $y$ is a $N \times 1$ measurement vector.
- $\Phi$ is $N \times M$ dictionary matrix where $M >> N$.
- $x$ is $M \times 1$ desired vector which is sparse with $k$ non-zero entries.
- $v$ is the measurement noise.
Extensions

- Block Sparsity
Extensions

- Block Sparsity
- Multiple Measurement Vectors (MMV)
Extensions

- Block Sparsity
- Multiple Measurement Vectors (MMV)
- Block MMV
- MMV with time varying sparsity
Multiple Measurement Vectors (MMV)

- Model

\[ Y_{N \times L} = \Phi_{N \times M} X_{M \times L} + V_{N \times L} \]

- Multiple measurements: \( L \) measurements
- Common Sparsity Profile: \( k \) nonzero rows
Applications

Signal Representation (Mallat, Coifman, Donoho, ..)

EEG/MEG (Leahy, Gorodnitsky, Ioannides, ..)

Robust Linear Regression and Outlier Detection (Jin, Giannakis, ..)

Speech Coding (Ozawa, Ono, Kroon, ..)

Compressed Sensing (Donoho, Candes, Tao, ..)

Magnetic Resonance Imaging (Lustig, ..)

Sparse Channel Equalization (Fevrier, Proakis, ..)

Face Recognition (Wright, Yang, ..)

Cognitive Radio (Eldar, ..)

and many more........
Applications

- Signal Representation (Mallat, Coifman, Donoho,..)
- EEG/MEG (Leahy, Gorodnitsky, Ioannides,..)
- Robust Linear Regression and Outlier Detection (Jin, Giannakis, ..)
- Speech Coding (Ozawa, Ono, Kroon,..)
- Compressed Sensing (Donoho, Candes, Tao,..)
- Magnetic Resonance Imaging (Lustig,..)
- Sparse Channel Equalization (Fevrier, Proakis,..)
- Face Recognition (Wright, Yang, ..)
- Cognitive Radio (Eldar, ..)

and many more........
Finding the Optimal Solution is NP hard. So need low complexity algorithms with reasonable performance.
Potential Algorithmic Approaches

Finding the Optimal Solution is NP hard. So need low complexity algorithms with reasonable performance.

**Greedy Search Techniques**

Matching Pursuit (MP), Orthogonal Matching Pursuit (OMP), ...
Potential Algorithmic Approaches

Finding the Optimal Solution is NP hard. So need low complexity algorithms with reasonable performance.

**Greedy Search Techniques**
- Matching Pursuit (MP), Orthogonal Matching Pursuit (OMP), ...

**Minimizing Diversity Measures (Regularization Framework)**
- Tractable Surrogate Cost functions: e.g. $\ell_1$ minimization, ...
Finding the Optimal Solution is NP hard. So need low complexity algorithms with reasonable performance.

**Greedy Search Techniques**

Matching Pursuit (MP), Orthogonal Matching Pursuit (OMP), ...

**Minimizing Diversity Measures (Regularization Framework)**

Tractable Surrogate Cost functions: e.g. $\ell_1$ minimization, ...

**Bayesian Methods**

Make appropriate Statistical assumptions on the solution (sparsity): **Choice of Prior**
Super Gaussian Distributions: Heavy tailed and sharper peak at origin compared to Gaussian.

Tractable representations using Scale Mixtures:
- Gaussian Scale Mixture (GSM)
- Laplacian Scale Mixture (LSM)
- Power Exponential Scale Mixture (PESM)
Gaussian Scale Mixtures

**Separability**: \( p(x) = \prod_i p(x_i) \)
Gaussian Scale Mixtures

Separability: \( p(x) = \prod_i p(x_i) \)

\[
p(x_i) = \int p(x_i | \gamma_i) p(\gamma_i) d\gamma_i = \int N(x_i; 0, \gamma_i) p(\gamma_i) d\gamma_i
\]
Gaussian Scale Mixtures

**Separability:** \( p(x) = \prod_i p(x_i) \)

\[
p(x_i) = \int p(x_i | \gamma_i) p(\gamma_i) d\gamma_i = \int N(x_i; 0, \gamma_i) p(\gamma_i) d\gamma_i
\]

**Theorem**

A density \( p(x) \) which is symmetric with respect origin, can be represented by a GSM iff \( p(\sqrt{x}) \) is completely monotonic on \((0, \infty)\).
**Separability:** \( p(x) = \prod_i p(x_i) \)

\[
p(x_i) = \int p(x_i|\gamma_i)p(\gamma_i)d\gamma_i = \int N(x_i;0,\gamma_i)p(\gamma_i)d\gamma_i
\]

**Theorem**

A density \( p(x) \) which is symmetric with respect origin, can be represented by a GSM iff \( p(\sqrt{x}) \) is completely monotonic on \((0, \infty)\).

Most of the sparse priors over \( x \) can be represented in this GSM form. [Palmer et al., 2006]
## Laplacian density

\[ p(x; a) = \frac{a}{2} \exp(-a|x|) \]

**Scale mixing density:** \( p(\gamma) = \frac{a^2}{2} \exp(-\frac{a^2}{2}\gamma), \gamma \geq 0. \)
### Laplacian density

\[ p(x; a) = \frac{a}{2} \exp(-a|x|) \]

**Scale mixing density:** \[ p(\gamma) = \frac{a^2}{2} \exp(-\frac{a^2}{2}\gamma), \gamma \geq 0. \]

### Student-t Distribution

\[ p(x; a, b) = \frac{b^a \Gamma(a + 1/2)}{(2\pi)^{0.5} \Gamma(a)} \frac{1}{(b + x^2/2)^{a+1/2}} \]

**Scale mixing density:** Gamma Distribution.
Examples of Gaussian Scale Mixture

Laplacian density

\[ p(x; a) = \frac{a}{2} \exp(-a|x|) \]

Scale mixing density: \( p(\gamma) = \frac{a^2}{2} \exp(-\frac{a^2}{2} \gamma), \gamma \geq 0. \)

Student-t Distribution

\[ p(x; a, b) = \frac{b^a \Gamma(a + 1/2)}{(2\pi)^{0.5} \Gamma(a)} \frac{1}{(b + x^2/2)^{a+1/2}} \]

Scale mixing density: Gamma Distribution.

Generalized Gaussian

\[ p(x; p) = \frac{1}{2\Gamma(1 + \frac{1}{p})} e^{-|x|^p} \]

Scale mixing density: Positive alpha stable density of order \( p/2. \)
GSM corresponds to $\ell_2$ norm based SSR algorithm.

LSM corresponds to $\ell_1$ norm based SSR algorithm.

Need a generalized scale mixture for a unified treatment of $\ell_1$ and $\ell_2$ minimization based SSR.
Power Exponential Scale Mixture Distributions (PESM)

**Power Exponential Distribution**

Also known as Box and Tiao (BT) or Generalized Gaussian distribution (GGD).

\[ p_{PE}(x; 0, \sigma, p) = Ke^{-\frac{|x|^p}{\sigma^p}} \]
Power Exponential Distribution

Also known as Box and Tiao (BT) or Generalized Gaussian distribution (GGD).

\[ p_{PE}(x; 0, \sigma, p) = Ke^{-\frac{|x|^p}{\sigma^p}} \]

Scale Mixture of Power Exponential:

\[ p(x_i) = \int p(x_i|\gamma_i)p(\gamma_i)d\gamma_i = \int p_{PE}(x_i; 0, \gamma_i, p)p(\gamma_i)d\gamma_i \]
Power Exponential Scale Mixture Distributions (PESM)

- Choice of $p=2$
  - Gaussian Scale Mixtures (GSM): $\ell_2$ norm minimization based algorithms.

- Choice of $p=1$
  - Laplacian Scale Mixtures (LSM): $\ell_1$ norm minimization based algorithms.

- PESM
  - Unified treatment of both $\ell_1$ and $\ell_2$ based algorithms.
PESM Example: Generalized t distribution

Inverse Generalized Gamma (GG) for scaling density:

\[
p(\gamma_i) = p_{GG}(\gamma_i; -p, \sigma, q) = \eta \left(\frac{\sigma}{\gamma_i}\right)^{pq} + 1 e^{-\left(\frac{\sigma}{\gamma_i}\right)^p} \]

A wide class of heavy tailed super gaussian densities can be represented by GT using suitable shape parameters \( p \) and \( q \).

Bhaskar D Rao  University of California, San Diego
Inverse Generalized Gamma (GG) for scaling density:

\[ p(\gamma_i) = p_{GG}(\gamma_i; -p, \sigma, q) = \eta (\sigma / \gamma_i)^{pq+1} e^{-(\sigma / \gamma_i)^p} \]
Inverse Generalized Gamma (GG) for scaling density:

\[ p(\gamma_i) = p_{GG}(\gamma_i; -p, \sigma, q) = \eta(\sigma/\gamma_i)^{pq+1} e^{-(\sigma/\gamma_i)^p} \]

\[ p_{GT}(x; \sigma, p, q) = K(1 + \frac{|x|^p}{q\sigma^p})^{-(q+1/p)} \]
Inverse Generalized Gamma (GG) for scaling density:

\[ p(\gamma_i) = p_{GG}(\gamma_i; -p, \sigma, q) = \eta (\sigma / \gamma_i)^{pq+1} e^{-(\sigma / \gamma_i)^p} \]

\[ p_{GT}(x; \sigma, p, q) = K (1 + \frac{|x|^p}{q\sigma^p})^{-(q+1/p)} \]

A wide class of heavy tailed super gaussian densities can be represented by GT using suitable shape parameters p and q.
### Table: Variants of Generalized t Distribution

<table>
<thead>
<tr>
<th>q</th>
<th>p</th>
<th>Distribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>$q \to \infty$</td>
<td>2</td>
<td>Normal</td>
</tr>
<tr>
<td>$q \to \infty$</td>
<td>1</td>
<td>Laplacian (Double Exponential)</td>
</tr>
<tr>
<td>$q \geq 0$ (degrees of freedom)</td>
<td>2</td>
<td>Student t distribution</td>
</tr>
<tr>
<td>$q \geq 0$ (shape parameter)</td>
<td>1</td>
<td>Generalized Double Pareto (GDP)</td>
</tr>
</tbody>
</table>
MAP Estimation (Type I)
MAP Estimation (Type I)

Hierarchical Bayes (Type II)
MAP Estimation Framework (Type I)

Problem Statement

\[ \hat{x} = \arg \max_x p(x|y) = \arg \max_x p(y|x)p(x) \]

Choice of \( p(x) = \frac{a}{2} e^{-a|x|} \) as Laplacian and Gaussian Likelihood assumption will lead to the familiar LASSO framework.
Hierarchical Bayesian Framework (Type II)

Problem Statement

\[ \gamma = \arg \max_\gamma p(\gamma | y) = \arg \max_\gamma p(y | \gamma) p(\gamma) \]

Using this estimate of \( \gamma \) we can compute our concerned posterior \( p(x | y; \hat{\gamma}) \).

Bhaskar D Rao  University of California, San Diego
Hierarchical Bayesian Framework (Type II)

Problem Statement

\[ \hat{\gamma} = \arg \max_{\gamma} p(\gamma|y) = \arg \max_{\gamma} p(y|\gamma)p(\gamma) \]

Using this estimate of \( \gamma \) we can compute our concerned posterior \( p(x|y; \hat{\gamma}) \).
Hierarchical Bayesian Framework (Type II)

Potential Advantages

- Averaging over $x$ leads to fewer minima in $p(\gamma|y)$.
- $\gamma$ can tie several parameters, leading to fewer parameters.
- Maximizing the true posterior mass over the subspaces spanned by non zero indexes instead of looking for the mode.

Bayesian LASSO

Let $p(x)$ be the Laplacian prior as GSM $a$: $p(x) = \int p(x|\gamma)p(\gamma)d\gamma = \int 1/\sqrt{2\pi\gamma} \exp(-x^2/2\gamma) \times a^2 \exp(-a^2/2\gamma) p(\gamma)d\gamma = a^2 \exp(-a|x|) a$.
Hierarchical Bayesian Framework (Type II)

Potential Advantages

- Averaging over $x$ leads to fewer minima in $p(\gamma|y)$.
- $\gamma$ can tie several parameters, leading to fewer parameters.
- Maximizing the true posterior mass over the subspaces spanned by non zero indexes instead of looking for the mode.

Bayesian LASSO

Laplacian $p(x)$ as GSM\(^a\):

$$p(x) = \int p(x|\gamma)p(\gamma)d\gamma$$

$$= \int \frac{1}{\sqrt{2\pi\gamma}} \exp\left(-\frac{x^2}{2\gamma}\right) \times \frac{a^2}{2} \exp\left(-\frac{a^2}{2\gamma}\right) \, d\gamma$$

$$= \frac{a}{2} \exp\left(-a|x|\right)$$

\(^a\) Bayesian Compressive Sensing Using Laplace Priors”, Babacan et al

Bhaskar D Rao  University of California, San Diego
Problem Statement

\[ \hat{x} = \arg \max_x \log p(x|y) = \arg \max_x \log p(y|x) + \log p(x) \]
Problem Statement

\[ \hat{x} = \arg \max_x \log p(x|y) = \arg \max_x \log p(y|x) + \log p(x) \]

Examples:

<table>
<thead>
<tr>
<th>Prior Distribution</th>
<th>Penalty Function</th>
<th>SSR Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal</td>
<td>|x|_2</td>
<td>Ridge Regression</td>
</tr>
<tr>
<td>Laplacian</td>
<td>|x|_1</td>
<td>LASSO</td>
</tr>
<tr>
<td>Student t distribution</td>
<td>\log(\epsilon + x^2)</td>
<td>Reweighted \ell_2 (Chartrand’s)</td>
</tr>
<tr>
<td>Generalized Double Pareto</td>
<td>\log(\epsilon +</td>
<td>x</td>
</tr>
</tbody>
</table>
### Problem Statement

\[
\hat{x} = \arg \max_x \log p(x|y) = \arg \max_x \log p(y|x) + \log p(x)
\]

### Examples:

<table>
<thead>
<tr>
<th>Prior Distribution</th>
<th>Penalty Function</th>
<th>SSR Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal</td>
<td>$</td>
<td></td>
</tr>
<tr>
<td>Laplacian</td>
<td>$</td>
<td></td>
</tr>
<tr>
<td>Student t distribution</td>
<td>$\log(\epsilon + x^2)$</td>
<td>Reweighted $\ell_2$ (Chartrand's)</td>
</tr>
<tr>
<td>Generalized Double Pareto</td>
<td>$\log(\epsilon +</td>
<td>x</td>
</tr>
</tbody>
</table>

PESM as sparsity promoting prior $p(x)$: Unified Type I Framework
Choice of Prior: $p(x)$

Any distribution in PESM class.
Unified Type I Framework

Choice of Prior: \( p(x) \)
Any distribution in PESM class.

EM Algorithm

- Complete Data Log-Likelihood:
  \[
  \log p(y, x, \gamma) = \log p(y|x) + \log p(x|\gamma) + \log p(\gamma)
  \]

- Hidden Variable: \( \gamma \)

- Concerned Posterior: \( p(\gamma|x, y) \sim p(\gamma|x) \) (From Markov chain).
Unified Type I: E step

\[ Q(\mathbf{x}) = \mathbb{E}_{\gamma|x} \left[ \log p(\mathbf{y}|\mathbf{x}) + \log p(\mathbf{x}|\gamma) + \log p(\gamma) \right] \]

E Step

- Only second term has dependencies on both \( \mathbf{x} \) and \( \gamma \).
- Compute \( E_{\gamma_i|x_i} \left[ \frac{1}{\gamma_i^p} \right] \)
Unified Type I: E step

\[ p'(x_i) = \frac{d}{dx_i} \int_0^\infty p(x_i | \gamma_i) p(\gamma_i) d\gamma_i \]

\[ = -p \times |x_i|^{p-1} \text{sign}(x_i) p(x_i) \int_0^\infty \frac{1}{\gamma_i^p} p(\gamma_i | x_i) d\gamma_i \]

\[ = -p \times |x_i|^{p-1} \text{sign}(x_i) p(x_i) E_{\gamma_i | x_i} \left[ \frac{1}{\gamma_i^p} \right] \]

E step

\[ E_{\gamma_i | x_i} \left[ \frac{1}{\gamma_i^p} \right] = - \frac{p'(x_i)}{p \times |x_i|^{p-1} \text{sign}(x_i) p(x_i)} \]
Unified Type I: E step

\[ p'(x_i) = \frac{d}{dx_i} \int_0^\infty p(x_i|\gamma_i)p(\gamma_i)d\gamma_i \]

\[ = -p \times |x_i|^{p-1}\text{sign}(x_i)p(x_i) \int_0^\infty \frac{1}{\gamma_i^p} p(\gamma_i|x_i) d\gamma_i \]

\[ = -p \times |x_i|^{p-1}\text{sign}(x_i)p(x_i)E_{\gamma_i|x_i}\left[\frac{1}{\gamma_i^p}\right] \]

E step

\[ E_{\gamma_i|x_i}\left[\frac{1}{\gamma_i^p}\right] = -\frac{p'(x_i)}{p \times |x_i|^{p-1}\text{sign}(x_i)p(x_i)} \]

Note: No need to know \( p(\gamma) \), as long as \( p(x) \) is known and has a PESM representation.
Unified Type I: M step

\[ \hat{x}^{(k+1)} = \arg \min_x \frac{1}{2\lambda} \|y - \Phi x\|^2 + \sum_i w_i^{(k)} |x_i|^p \]

Where,

\[ w_i^{(k)} = E_{\gamma_i|x_i^{(k)}} \left[ \frac{1}{\gamma_i^p} \right] \]

Special Case: Generalized t distribution

\[ w_i^{(k)} = \frac{q + 1/p}{q\sigma + |x_i^{(k)}|^p} \]
Hierarchical Bayesian Framework (Type II)

Estimate of the posterior distribution for $x$ using estimated $\hat{\gamma}$; i.e. $p(x|y; \hat{\gamma})$.

Choice of GSM as $p(x)$ leads to Sparse Bayesian Learning.
Hierarchical Bayesian Framework (Type II)

Estimate of the posterior distribution for $x$ using estimated $\hat{\gamma}$; i.e. $p(x|y; \hat{\gamma})$.

Choice of GSM as $p(x)$ leads to Sparse Bayesian Learning.
Sparse Bayesian Learning (Type II)

\[ y = \Phi x + v \]

Solving for MAP estimate of \( \hat{\gamma} \)

\[ \hat{\gamma} = \arg \max_{\gamma} p(\gamma | y) = \arg \max_{\gamma} p(y | \gamma) p(\gamma) \]

What is \( p(y | \gamma) \)?

Given \( \gamma \), \( x \) is Gaussian with mean zero and Covariance matrix \( \Gamma \) with \( \Gamma = \text{diag}(\gamma) \), i.e.

\[ p(x | \gamma) = \mathcal{N}(x; 0, \Gamma) = \prod_{i} \mathcal{N}(x_i; 0, \gamma_i) \]

Then \( p(y | \gamma) = \mathcal{N}(y; 0, \Sigma_y) \), where \( \Sigma_y = \sigma^2 I + \Phi \Gamma \Phi^T \)

\[ p(y | \gamma) = \frac{1}{\sqrt{(2\pi)^N \sigma^2}} e^{-\frac{1}{2}(y^T \Sigma_y^{-1} y)} \]
Sparse Bayesian Learning (Type II)

\[ y = \Phi x + v \]
Sparse Bayesian Learning (Type II)

\[ y = \Phi x + v \]

Solving for MAP estimate of \( \gamma \)

\[ \hat{\gamma} = \arg \max_{\gamma} p(\gamma|y) = \arg \max_{\gamma} p(y|\gamma)p(\gamma) \]

Given \( \gamma \), \( x \) is Gaussian with mean zero and Covariance matrix \( \Gamma \) with \( \Gamma = \text{diag}(\gamma) \), i.e.

\[ p(x|\gamma) = \mathcal{N}(x; 0, \Gamma) = \prod \mathcal{N}(x_i; 0, \gamma_i) \]

Then \( p(y|\gamma) = \mathcal{N}(y; 0, \Sigma_y) \), where \( \Sigma_y = \sigma^2 I + \Phi \Gamma \Phi^T \),

\[ p(y|\gamma) = \frac{1}{\sqrt{(2\pi)^N \Sigma_y}} e^{-\frac{1}{2} y^T \Sigma_y^{-1} y} \]
Sparse Bayesian Learning (Type II)

\[ y = \Phi x + v \]

Solving for MAP estimate of \( \gamma \)

\[ \hat{\gamma} = \arg \max_{\gamma} p(\gamma | y) = \arg \max_{\gamma} p(y | \gamma) p(\gamma) \]

What is \( p(y | \gamma) \)

Given \( \gamma \), \( x \) is Gaussian with mean zero and covariance matrix \( \Gamma \) with \( \Gamma = \text{diag}(\gamma) \), i.e.

\[ p(x | \gamma) = \prod_{i} \mathcal{N}(x_i; 0, \gamma_i) \]

Then

\[ p(y | \gamma) = \frac{1}{\sqrt{(2\pi)^N \det(\Sigma_y)}} e^{-\frac{1}{2} y^T \Sigma_y^{-1} y} \]

where \( \Sigma_y = \sigma^2 I + \Phi \Gamma \Phi^T \).
\( y = \Phi x + v \)

**Solving for MAP estimate of \( \gamma \)**

\[
\hat{\gamma} = \arg \max_{\gamma} p(\gamma|y) = \arg \max_{\gamma} p(y|\gamma)p(\gamma)
\]

**What is \( p(y|\gamma) \)**

Given \( \gamma \), \( x \) is Gaussian with mean zero and Covariance matrix \( \Gamma \) with \( \Gamma = \text{diag}(\gamma) \), i.e. \( p(x|\gamma) = N(x; 0, \Gamma) = \prod_i N(x_i; 0, \gamma_i) \).
\[ y = \Phi x + v \]

**Solving for MAP estimate of \( \gamma \)**

\[
\hat{\gamma} = \arg \max_{\gamma} p(\gamma|y) = \arg \max_{\gamma} p(y|\gamma)p(\gamma)
\]

**What is \( p(y|\gamma) \)**

Given \( \gamma \), \( x \) is Gaussian with mean zero and Covariance matrix \( \Gamma \) with \( \Gamma = \text{diag}(\gamma) \), i.e. \( p(x|\gamma) = N(x; 0, \Gamma) = \prod N(x_i; 0, \gamma_i) \).

Then \( p(y|\gamma) = N(y; 0, \Sigma_y) \), where \( \Sigma_y = \sigma^2 I + \Phi \Gamma \Phi^T \),

\[
p(y|\gamma) = \frac{1}{\sqrt{(2\pi)^N|\Sigma_y|}} e^{-\frac{1}{2} y^T \Sigma_y^{-1} y}
\]
Sparse Bayesian Learning (Tipping)

\[ y = \Phi x + v \]

Solving for the optimal \( \gamma \)

\[ \hat{\gamma} = \arg \max_{\gamma} p(\gamma|y) = \arg \max_{\gamma} p(y|\gamma)p(\gamma) \]

\[ = \arg \min_{\gamma} \log |\Sigma_y| + y^T \Sigma_y^{-1} y - 2 \sum_i \log p(\gamma_i) \]

where, \( \Sigma_y = \sigma^2 I + \Phi \Gamma \Phi^T \) and \( \Gamma = \text{diag}(\gamma) \)
Sparse Bayesian Learning (Tipping)

\[ y = \Phi x + \nu \]

Solving for the optimal \( \gamma \)

\[ \hat{\gamma} = \arg \max_{\gamma} p(\gamma | y) = \arg \max_{\gamma} p(y | \gamma) p(\gamma) \]
\[ = \arg \min_{\gamma} \log |\Sigma_y| + y^T \Sigma_y^{-1} y - 2 \sum_i \log p(\gamma_i) \]

where, \( \Sigma_y = \sigma^2 I + \Phi \Gamma \Phi^T \) and \( \Gamma = \text{diag}(\gamma) \)

Computational Methods

Many options for solving the above optimization problem, e.g. Majorization Minimization, Expectation-Maximization (EM).
Sparse Bayesian Learning

\[ y = \Phi x + v \]

**Computing Posterior**

Now because of our convenient GSM choice, posterior can be easily computed, i.e, \( p(x|y; \hat{\gamma}) = N(\mu_x, \Sigma_x) \) where,

\[ \mu_x = E[x|y; \hat{\gamma}] = \hat{\Gamma} \Phi^T (\sigma^2 I + \Phi \hat{\Gamma} \Phi^T)^{-1} y \]

\[ \Sigma_x = \text{Cov}[x|y; \hat{\gamma}] = \hat{\Gamma} - \hat{\Gamma} \Phi^T (\sigma^2 I + \Phi \hat{\Gamma} \Phi^T)^{-1} \Phi \hat{\Gamma} \]
\[ y = \Phi x + v \]

**Computing Posterior**

Now because of our convenient GSM choice, posterior can be easily computed, i.e, \( p(x|y; \hat{\gamma}) = N(\mu_x, \Sigma_x) \) where,

\[
\mu_x = E[x|y; \hat{\gamma}] = \hat{\Gamma} \Phi^T (\sigma^2 I + \Phi \hat{\Gamma} \Phi^T)^{-1} y
\]

\[
\Sigma_x = Cov[x|y; \hat{\gamma}] = \hat{\Gamma} - \hat{\Gamma} \Phi^T (\sigma^2 I + \Phi \hat{\Gamma} \Phi^T)^{-1} \Phi \hat{\Gamma}
\]

\( \mu_x \) can be used as a point estimate.
\[ \mathbf{y} = \Phi \mathbf{x} + \mathbf{v} \]

**Computing Posterior**

Now because of our convenient GSM choice, posterior can be easily computed, i.e., \( p(\mathbf{x}|\mathbf{y}; \hat{\gamma}) = N(\mu_\mathbf{x}, \Sigma_\mathbf{x}) \) where,

\[
\mu_\mathbf{x} = E[\mathbf{x}|\mathbf{y}; \hat{\gamma}] = \hat{\Gamma} \Phi^T (\sigma^2 I + \Phi \hat{\Gamma} \Phi^T)^{-1} \mathbf{y}
\]

\[
\Sigma_\mathbf{x} = \text{Cov}[\mathbf{x}|\mathbf{y}; \hat{\gamma}] = \hat{\Gamma} - \hat{\Gamma} \Phi^T (\sigma^2 I + \Phi \hat{\Gamma} \Phi^T)^{-1} \Phi \hat{\Gamma}
\]

\( \mu_\mathbf{x} \) can be used as a point estimate.

Sparsity of \( \mu_\mathbf{x} \) is achieved through sparsity in \( \gamma \).
\[ y = \Phi x + v \]

**Computing Posterior**

Now because of our convenient GSM choice, posterior can be easily computed, i.e, \( p(x|y; \hat{\gamma}) = N(\mu_x, \Sigma_x) \) where,

\[
\mu_x = E[x|y; \hat{\gamma}] = \hat{\Gamma} \Phi^T (\sigma^2 I + \Phi \hat{\Gamma} \Phi^T)^{-1} y
\]
\[
\Sigma_x = \text{Cov}[x|y; \hat{\gamma}] = \hat{\Gamma} - \hat{\Gamma} \Phi^T (\sigma^2 I + \Phi \hat{\Gamma} \Phi^T)^{-1} \Phi \hat{\Gamma}
\]

\( \mu_x \) can be used as a point estimate.

Sparsity of \( \mu_x \) is achieved through sparsity in \( \gamma \).

Another parameter of interest for the EM algorithm

\[
E(x_i^2|y, \hat{\gamma}) = \mu_x^2(i) + \Sigma_x(i, i)
\]
EM algorithm: Updating $\gamma$

Treating $(y, x)$ as complete data and $x$ as hidden variable.

$$\log p(y, x, \gamma) = \log p(y|x) + \log p(x|\gamma) + \log p(\gamma)$$

**E step**

$$Q(\gamma|\gamma_k) = E_{x|y; \gamma_k} \left[ \log p(y|x) + \log p(x|\gamma) + \log p(\gamma) \right]$$

**M step**

$$\gamma_{k+1} = \arg\max_{\gamma} \sum_{i=1}^M \left[ \frac{1}{2} \left( x^2_i + \frac{1}{2} \log \gamma_i \right) - \log p(\gamma_i) \right]$$

Solving this optimization problem with a non-informative prior $p(\gamma)$,

$$\gamma_{k+1} = E(x^2_i|y, \gamma_k) = \mu_x(x_i^2) + \sum_{x_i} x_i$$

Bhaskar D Rao
University of California, San Diego
EM algorithm: Updating $\gamma$

Treating $(y, x)$ as complete data and vector $x$ as hidden variable.

$$\log p(y, x, \gamma) = \log p(y|x) + \log p(x|\gamma) + \log p(\gamma)$$
EM algorithm: Updating $\gamma$

Treating $(y, x)$ as complete data and vector $x$ as hidden variable.

$$\log p(y, x, \gamma) = \log p(y|x) + \log p(x|\gamma) + \log p(\gamma)$$

**E step**

$$Q(\gamma|\gamma^k) = \mathbb{E}_{x|y, \gamma^k}[\log p(y|x) + \log p(x|\gamma) + \log p(\gamma)]$$
EM algorithm: Updating $\gamma$

Treating $(y, x)$ as complete data and vector $x$ as hidden variable.

$$\log p(y, x, \gamma) = \log p(y|x) + \log p(x|\gamma) + \log p(\gamma)$$

**E step**

$$Q(\gamma|\gamma^k) = \mathbb{E}_{x|y;\gamma^k}[\log p(y|x) + \log p(x|\gamma) + \log p(\gamma)]$$

**M step**

$$\gamma^{k+1} = \arg\max_{\gamma} Q(\gamma|\gamma^k) = \arg\max_{\gamma} \mathbb{E}_{x|y;\gamma^k}[\log p(x|\gamma) + \log p(\gamma)]$$

$$= \arg\min_{\gamma} \mathbb{E}_{x|y;\gamma^k} \sum_{i=1}^{M} \left[ \left( \frac{x_i^2}{2\gamma_i} + \frac{1}{2} \log \gamma_i \right) - \log p(\gamma_i) \right]$$
EM algorithm: Updating $\gamma$

Treating $(\mathbf{y}, \mathbf{x})$ as complete data and vector $\mathbf{x}$ as hidden variable.

$$\log p(y, x, \gamma) = \log p(y|x) + \log p(x|\gamma) + \log p(\gamma)$$

**E step**

$$Q(\gamma|\gamma^k) = \mathbb{E}_{\mathbf{x}|y;\gamma^k} \left[ \log p(y|x) + \log p(x|\gamma) + \log p(\gamma) \right]$$

**M step**

\begin{align*}
\gamma^{k+1} &= \arg\max_\gamma Q(\gamma|\gamma^k) = \arg\max_\gamma \mathbb{E}_{\mathbf{x}|y;\gamma^k} \left[ \log p(x|\gamma) + \log p(\gamma) \right] \\
&= \arg\min_\gamma \mathbb{E}_{\mathbf{x}|y;\gamma^k} \sum_{i=1}^M \left[ \left( \frac{x_i^2}{2\gamma_i} + \frac{1}{2} \log \gamma_i \right) - \log p(\gamma_i) \right]
\end{align*}

Solving this optimization problem with a non-informative prior $p(\gamma)$,

$$\gamma_{i}^{k+1} = E(x_i^2|\mathbf{y}, \gamma^k) = \mu_x(i)^2 + \Sigma_x(i, i)$$
Type II (SBL) properties

Local minima are sparse, i.e. have at most \( N \) nonzero \( \gamma_i \).

Cost function \( p(\gamma | y) \) is generally much smoother than the associated MAP estimation objective \( p(x | y) \). Fewer local minima.

In high signal to noise ratio, the global minima is the sparsest solution. No structural problems.

Attempts to approximate the posterior distribution \( p(x | y) \) in the area with significant mass.
Type II (SBL) properties

- Local minima are sparse, i.e. have at most $N$ nonzero $\gamma_i$
Local minima are sparse, i.e. have at most $N$ nonzero $\gamma_i$

Cost function $p(\gamma|y)$ is generally much smoother than the associated MAP estimation objective $p(x|y)$. Fewer local minima.
Type II (SBL) properties

- Local minima are sparse, i.e. have at most $N$ nonzero $\gamma_i$
- Cost function $p(\gamma|y)$ is generally much smoother than the associated MAP estimation objective $p(x|y)$. Fewer local minima.
- In high signal to noise ratio, the global minima is the sparsest solution. No structural problems.
Type II (SBL) properties

- Local minima are sparse, i.e. have at most $N$ nonzero $\gamma_i$
- Cost function $p(\gamma|y)$ is generally much smoother than the associated MAP estimation objective $p(x|y)$. Fewer local minima.
- In high signal to noise ratio, the global minima is the sparsest solution. No structural problems.
- Attempts to approximate the posterior distribution $p(x|y)$ in the area with significant mass.
Algorithmic Variants

- Fixed Point iteration based on setting the derivative of the objective function to zero (Tipping)
- Sequential search for the significant $\gamma$'s (Tipping and Faul)
- Majorization-Minimization based approach (Wipf and Nagarajan)
- Reweighted $\ell_1$ and $\ell_2$ algorithms (Wipf and Nagarajan)
- Approximate Message Passing (AlShoukairi and Rao)
Algorithmic Variants

- Fixed Point iteration based on setting the derivative of the objective function to zero (Tipping)

- Sequential search for the significant $\gamma$'s (Tipping and Faul)

- Majorization-Minimization based approach (Wipf and Nagarajan)

- Reweighted $\ell_1$ and $\ell_2$ algorithms (Wipf and Nagarajan)

- Approximate Message Passing (AlShoukairi and Rao)
Algorithmic Variants

- Fixed Point iteration based on setting the derivative of the objective function to zero (Tipping)
- Sequential search for the significant $\gamma$'s (Tipping and Faul)

Reweighted $\ell_1$ and $\ell_2$ algorithms (Wipf and Nagarajan)

Approximate Message Passing (AlShoukairi and Rao)
Algorithmic Variants

- Fixed Point iteration based on setting the derivative of the objective function to zero (Tipping)
- Sequential search for the significant $\gamma$’s (Tipping and Faul)
- Majorization-Minimization based approach (Wipf and Nagarajan)
Algorithmic Variants

- Fixed Point iteration based on setting the derivative of the objective function to zero (Tipping)
- Sequential search for the significant $\gamma$’s (Tipping and Faul)
- Majorization-Minimization based approach (Wipf and Nagarajan)
- Reweighted $\ell_1$ and $\ell_2$ algorithms (Wipf and Nagarajan)
Algorithmic Variants

- Fixed Point iteration based on setting the derivative of the objective function to zero (Tipping)
- Sequential search for the significant $\gamma$’s (Tipping and Faul)
- Majorization-Minimization based approach (Wipf and Nagarajan)
- Reweighted $\ell_1$ and $\ell_2$ algorithms (Wipf and Nagarajan)
- Approximate Message Passing (AlShoukairi and Rao)
In E step we need to compute the conditional expectation.
Closed form may not be available depending on the choice of p (distributional parameter of PESM).
Alternative: MCMC technique.
Type II using PESM

- In E step we need to compute the conditional expectation.
- Closed form may not be available depending on the choice of \( p \) (distributional parameter of PESM).
- Alternative: MCMC technique.

LSM

Using the fact that a Laplacian density has a GSM representation, a tractable 3 layer hierarchical model can be developed.
Simulation Results

Parameters

1. \( N = 50, \ M = 250. \)

2. Dictionary Elements: Normal Distribution with mean = 0 and standard deviation = 1.

3. Distribution of non zero elements
   (I) Zero mean unit variance Gaussian.
   (II) Student t distribution with degrees of freedom \( \nu = 3 \).
   (Super-Gaussian)
   (III) Uniform \( \pm 1 \) random spikes.
Simulation Results: Gaussian

Figure: Recovery performance with Gaussian distributed non-zero coefficients
Simulation Results: Super Gaussian

Figure: Recovery performance with Super Gaussian (Student t) distributed non zero coefficients
Figure: Recovery performance with uniform spikes as non zero coefficients
**Model**

\[ Y_{N \times L} = \Phi_{N \times M} X_{M \times L} + V_{N \times L} \]

- Multiple measurements: \( L \) measurements
- Common Sparsity Profile: \( k \) nonzero rows

\( k \ll M \)
Bayesian Methods: GSM Extension

\[ X = \gamma G \]

where, 

\[ G \sim \mathcal{N}(g; 0, B) \]

\[ \gamma \] is a positive random variable, which is independent of \( G \).

\[ p(x) = \int p(x|\gamma)p(\gamma)d\gamma = \int \mathcal{N}(x; 0, \gamma B)p(\gamma)d\gamma \]

\[ B = I \text{ if the row entries are assumed independent.} \]

One \( \gamma \) per row vector. Complexity of estimating \( \gamma \) does not grow with \( L \).

The EM algorithm is also very tractable.

Bhaskar D Rao  University of California, San Diego
Representation for Random Vectors (Rows for MMV)

$$\mathbf{X} = \gamma \mathbf{G} \text{ where, } \mathbf{G} \sim N(\mathbf{g}; 0, \mathbf{B})$$

$\gamma$ is a positive random variable, which is independent of $\mathbf{G}$.
Bayesian Methods: GSM Extension

Representation for Random Vectors (Rows for MMV)

\[ X = \gamma G \text{ where, } G \sim N(g; 0, B) \]

\( \gamma \) is a positive random variable, which is independent of \( G \).

\[
p(x) = \int p(x|\gamma)p(\gamma)d\gamma = \int N(x; 0, \gamma B)p(\gamma)d\gamma
\]
Representation for Random Vectors (Rows for MMV)

\[ \mathbf{X} = \gamma \mathbf{G} \text{ where, } \mathbf{G} \sim N(\mathbf{g}; 0, \mathbf{B}) \]

\( \gamma \) is a positive random variable, which is independent of \( \mathbf{G} \).

\[
p(\mathbf{x}) = \int p(\mathbf{x} | \gamma)p(\gamma)d\gamma = \int N(\mathbf{x}; 0, \gamma \mathbf{B})p(\gamma)d\gamma
\]

- \( \mathbf{B} = \mathbf{I} \) if the row entries are assumed independent.
Representation for Random Vectors (Rows for MMV)

\[ \mathbf{X} = \gamma \mathbf{G} \text{ where, } \mathbf{G} \sim \mathcal{N}(\mathbf{g}; 0, \mathbf{B}) \]

\( \gamma \) is a positive random variable, which is independent of \( \mathbf{G} \).

\[ p(\mathbf{x}) = \int p(\mathbf{x}|\gamma)p(\gamma)d\gamma = \int \mathcal{N}(\mathbf{x}; 0, \gamma\mathbf{B})p(\gamma)d\gamma \]

- \( \mathbf{B} = \mathbf{I} \) if the row entries are assumed independent.
- One \( \gamma \) per row vector. Complexity of estimating \( \gamma \) does not grow with \( L \).
Representation for Random Vectors (Rows for MMV)

$$\mathbf{X} = \gamma \mathbf{G} \text{ where, } \mathbf{G} \sim N(\mathbf{g}; 0, \mathbf{B})$$

$\gamma$ is a positive random variable, which is independent of $\mathbf{G}$.

$$p(\mathbf{x}) = \int p(\mathbf{x}|\gamma)p(\gamma)d\gamma = \int N(\mathbf{x}; 0, \gamma \mathbf{B})p(\gamma)d\gamma$$

- $\mathbf{B} = \mathbf{I}$ if the row entries are assumed independent.
- One $\gamma$ per row vector. Complexity of estimating $\gamma$ does not grow with $L$.
- The EM algorithm is also very tractable.
MMV Empirical Comparison: 1000 trials

N=50, M=250, L=3 (No noise)
Sparse Signal Recovery (SSR) and Compressed Sensing (CS) are interesting new signal processing tools with many potential applications. Many algorithmic options exist to solve the underlying sparse signal recovery problem; Greedy Search Techniques, regularization methods, Bayesian methods, among others. Bayesian methods offer interesting algorithmic options to the Sparse Signal Recovery problem. MAP methods (reweighted $\ell_1$ and $\ell_2$ methods), Hierarchical Bayesian Methods (Sparse Bayesian Learning) are versatile and can be more easily employed in problems with structure. Algorithms can often be justified by studying the resulting objective functions.
Sparse Signal Recovery (SSR) and Compressed Sensing (CS) are interesting new signal processing tools with many potential applications.
Sparse Signal Recovery (SSR) and Compressed Sensing (CS) are interesting new signal processing tools with many potential applications.

Many algorithmic options exist to solve the underlying sparse signal recovery problem; Greedy Search Techniques, regularization methods, Bayesian methods, among others.
Sparse Signal Recovery (SSR) and Compressed Sensing (CS) are interesting new signal processing tools with many potential applications.

Many algorithmic options exist to solve the underlying sparse signal recovery problem; Greedy Search Techniques, regularization methods, Bayesian methods, among others.

Bayesian methods offer interesting algorithmic options to the Sparse Signal Recovery problem.
**Summary**

- Sparse Signal Recovery (SSR) and Compressed Sensing (CS) are interesting new signal processing tools with many potential applications.

- Many algorithmic options exist to solve the underlying sparse signal recovery problem; Greedy Search Techniques, regularization methods, Bayesian methods, among others.

- Bayesian methods offer interesting algorithmic options to the Sparse Signal Recovery problem
  - MAP methods (reweighted $\ell_1$ and $\ell_2$ methods)
Sparse Signal Recovery (SSR) and Compressed Sensing (CS) are interesting new signal processing tools with many potential applications.

Many algorithmic options exist to solve the underlying sparse signal recovery problem; Greedy Search Techniques, regularization methods, Bayesian methods, among others.

Bayesian methods offer interesting algorithmic options to the Sparse Signal Recovery problem

- MAP methods (reweighted $\ell_1$ and $\ell_2$ methods)
- Hierarchical Bayesian Methods (Sparse Bayesian Learning)
Sparse Signal Recovery (SSR) and Compressed Sensing (CS) are interesting new signal processing tools with many potential applications.

Many algorithmic options exist to solve the underlying sparse signal recovery problem; Greedy Search Techniques, regularization methods, Bayesian methods, among others.

Bayesian methods offer interesting algorithmic options to the Sparse Signal Recovery problem

- MAP methods (reweighted $\ell_1$ and $\ell_2$ methods)
- Hierarchical Bayesian Methods (Sparse Bayesian Learning)
- Versatile and can be more easily employed in problems with structure
Sparse Signal Recovery (SSR) and Compressed Sensing (CS) are interesting new signal processing tools with many potential applications.

Many algorithmic options exist to solve the underlying sparse signal recovery problem; Greedy Search Techniques, regularization methods, Bayesian methods, among others.

Bayesian methods offer interesting algorithmic options to the Sparse Signal Recovery problem

- MAP methods (reweighted $\ell_1$ and $\ell_2$ methods)
- Hierarchical Bayesian Methods (Sparse Bayesian Learning)
- Versatile and can be more easily employed in problems with structure
- Algorithms can often be justified by studying the resulting objective functions.