Bayesian Methods for Sparse Signal Recovery

Bhaskar D Rao[1]
University of California, San Diego

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

Motivation

Motivation

Sparse Signal Recovery is an interesting area with many potential applications.

Motivation

Sparse Signal Recovery is an interesting area with many potential applications.

Methods developed for solving sparse signal recovery problem can be a valuable tool for signal processing practitioners.

Motivation

Sparse Signal Recovery is an interesting area with many potential applications.

Methods developed for solving sparse signal recovery problem can be a valuable tool for signal processing practitioners.

Many interesting developments in recent past that make the subject timely.

Motivation

Sparse Signal Recovery is an interesting area with many potential applications.

Methods developed for solving sparse signal recovery problem can be a valuable tool for signal processing practitioners.

Many interesting developments in recent past that make the subject timely.

Bayesian Framework offers some interesting options.

Outline

Outline

- Sparse Signal Recovery (SSR) Problem and some Extensions

Outline

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

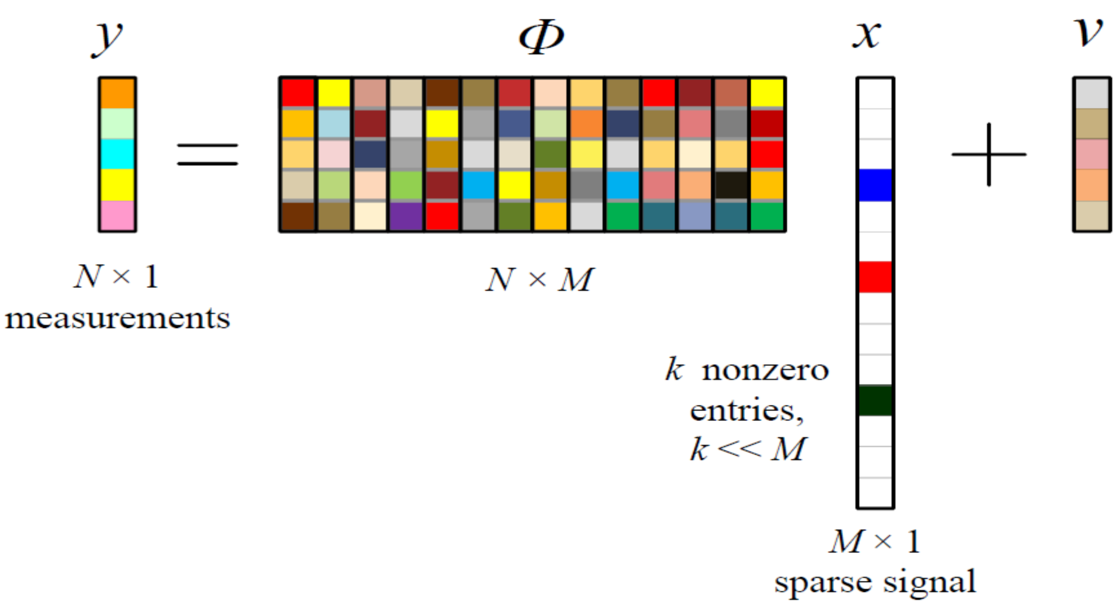
Outline

- Sparse Signal Recovery (SSR) Problem and some Extensions
- Applications
- Bayesian Methods
  - MAP estimation
  - Empirical Bayes

Outline

- Sparse Signal Recovery (SSR) Problem and some Extensions
- Applications
- Bayesian Methods
    - MAP estimation
    - Empirical Bayes
- Summary

Problem Description: Sparse Signal Recovery (SSR)



- y is a $N \times 1$ measurement vector.
- Φ 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.

Problem Statement: SSR

Problem Statement: SSR

Noise Free Case

Given a target signal y and dictionary Φ, find the weights x that solve,

$$\min_x \sum_i I(x_i \neq 0) \ \text{ subject to } \ y = \Phi x$$

$I(.)$ is the indicator function.

Problem Statement: SSR

Noise Free Case

Given a target signal y and dictionary Φ, find the weights x that solve,

$$\min_x \sum_i I(x_i \neq 0) \text{ subject to } y = \Phi x$$

$I(.)$ is the indicator function.

Noisy case

Given a target signal y and dictionary Φ, find the weights x that solve,

$$\min_x \sum_i I(x_i \neq 0) \text{ subject to } \|y - \Phi x\|_2 < \beta$$

Useful Extensions

Useful Extensions

- Block Sparsity

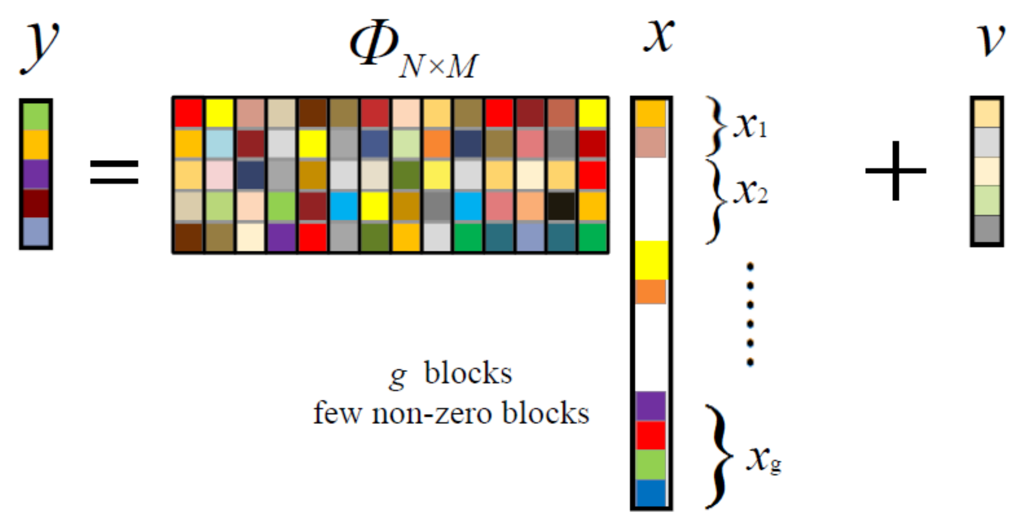
Useful Extensions

- Block Sparsity
- Multiple Measurement Vectors (MMV)

Useful Extensions

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

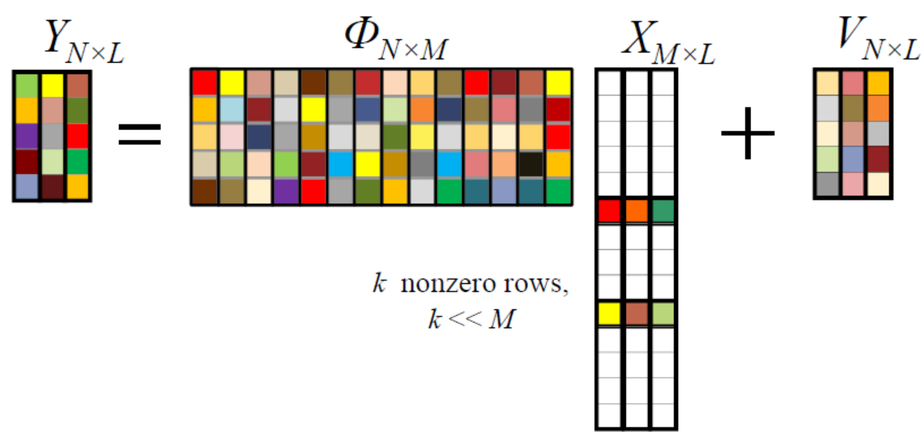
Block Sparsity



Variations include equal blocks, unequal blocks, block boundary known or unknown.

Multiple Measurement Vectors (MMV)

- Model



- Multiple measurements: L measurements
- Common Sparsity Profile: k nonzero rows

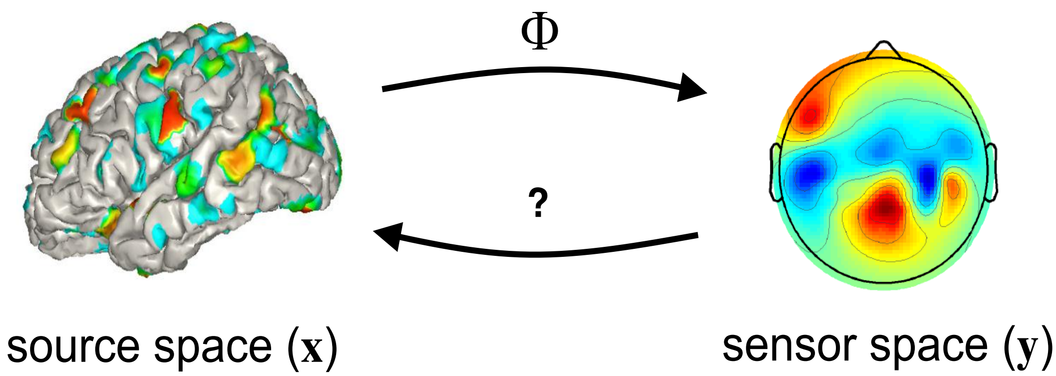
Applications

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.........

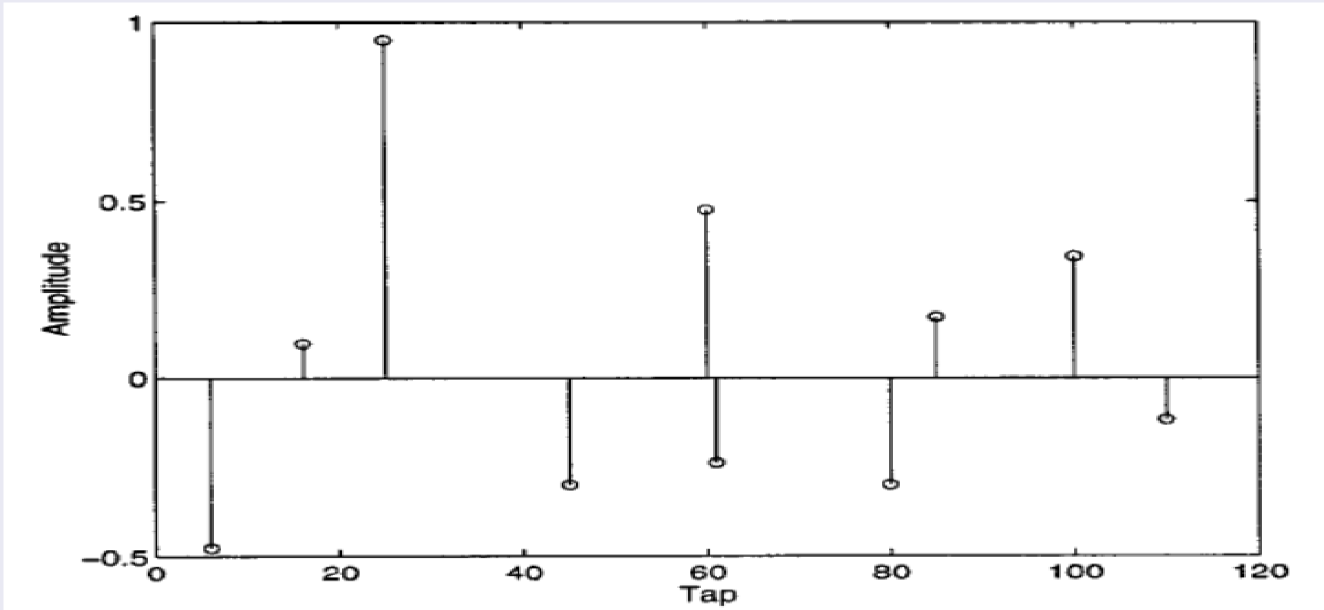
MEG/EEG Source Localization



- Forward model dictionary Φ can be computed using Maxwell's equations [Sarvas,1987].
- In many situations the active brain regions may be relatively sparse, and so solving a sparse inverse problem is required.

[Baillet et al., 2001]

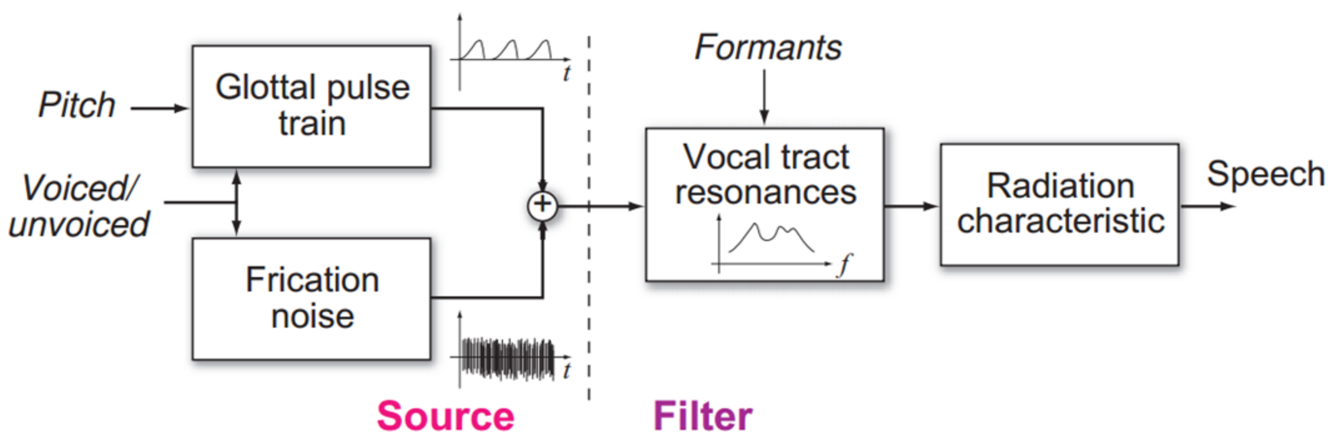
Sparse Channel Estimation



Potential Application: Underwater Acoustics

Speech Modeling and Deconvolution

Speech Model

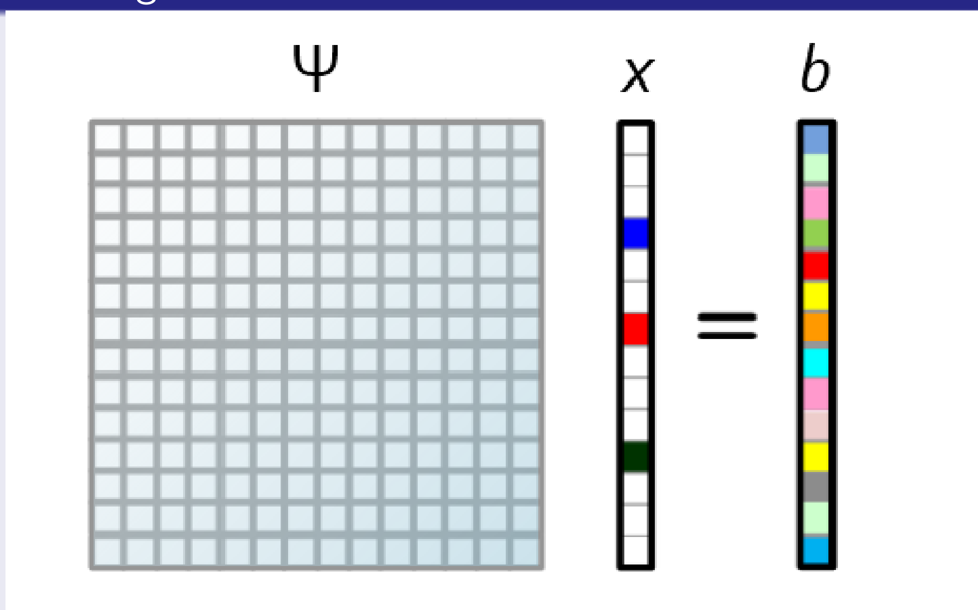


Speech specific assumptions: Voiced excitation is block sparse and the filter is an all pole filter $\frac{1}{A(z)}$

Compressive Sampling (CS)

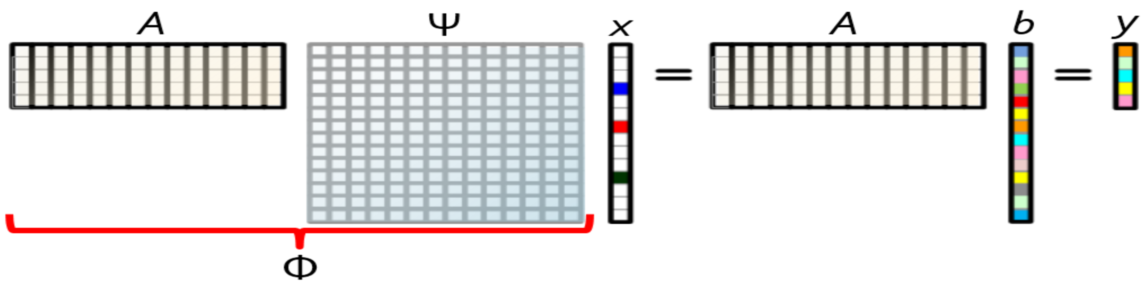
Compressive Sampling (CS)

Transform Coding

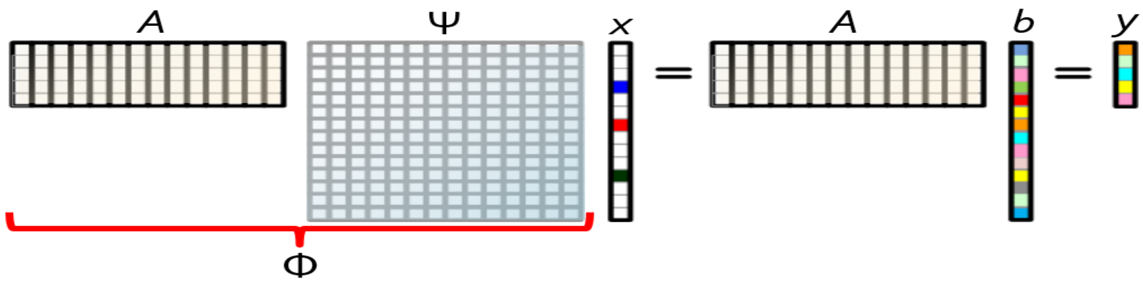


Ψ is the transform and b is the original data/image.

Compressive Sampling (CS)



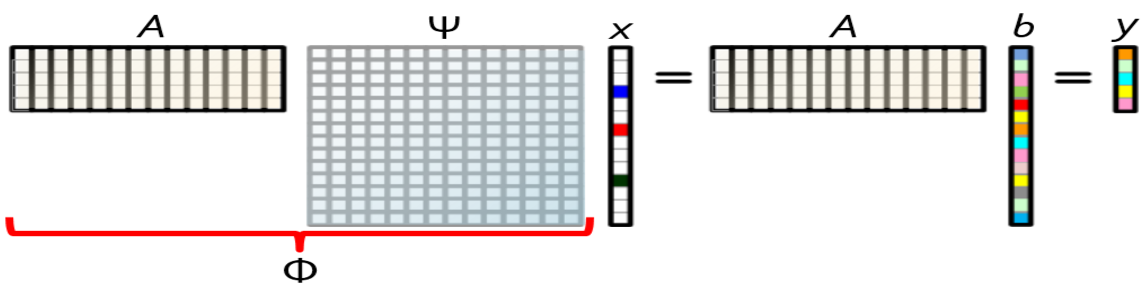
Compressive Sampling (CS)



Computation:

- Solve for x such that $\Phi x = y$.
- Reconstruction: $b = \Psi x$

Compressive Sampling (CS)



Computation:

- Solve for x such that $\Phi x = y$.
- Reconstruction: $b = \Psi x$

Issues:

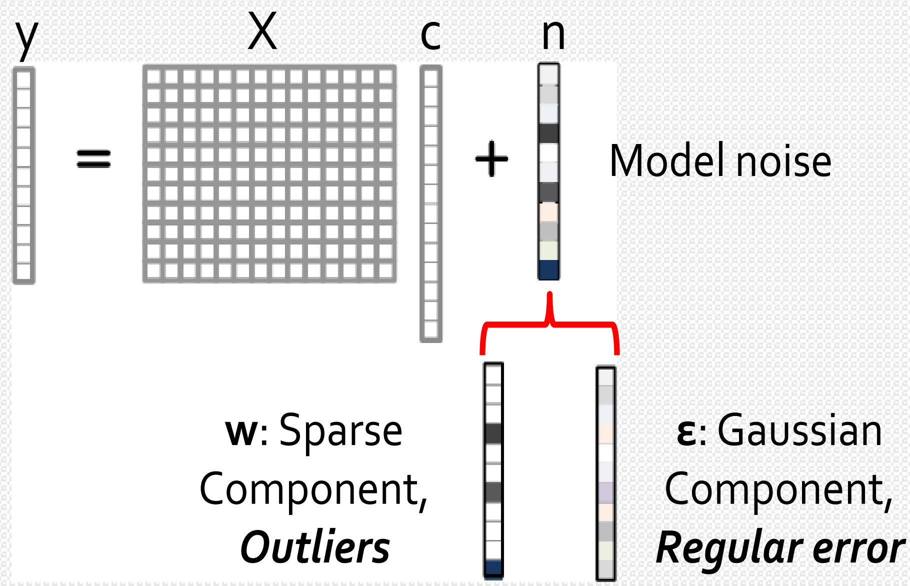
- Need to recover sparse signal x with constraint $\Phi x = y$.
- Need to design sampling matrix A.

Robust Linear Regression

X, y: data;

c: regression coeffs.;

n: model noise;



Transform into overcomplete representation:

$$Y = Xc + \Phi w + \varepsilon, \text{ where } \Phi = I,$$

or

$$Y = [X, \Phi]\begin{bmatrix} c \\ w \end{bmatrix} + \varepsilon$$

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

Indicator function is not continuous. Define Surrogate Cost functions that are more tractable and whose minimization leads to sparse solutions, e.g. ℓ_1 minimization.

Bayesian Methods

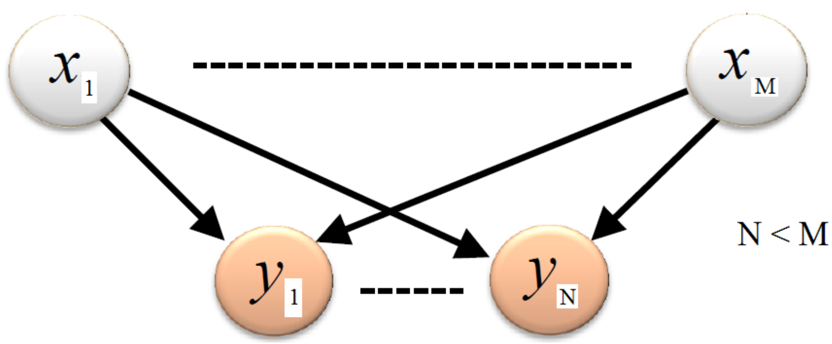
Make appropriate Statistical assumptions on the solution and apply estimation techniques to identify the desired sparse solution.

Bayesian Methods

1. MAP Estimation Framework (Type I)

2. Hierarchical Bayesian Framework (Type II)

MAP Estimation Framework (Type I)

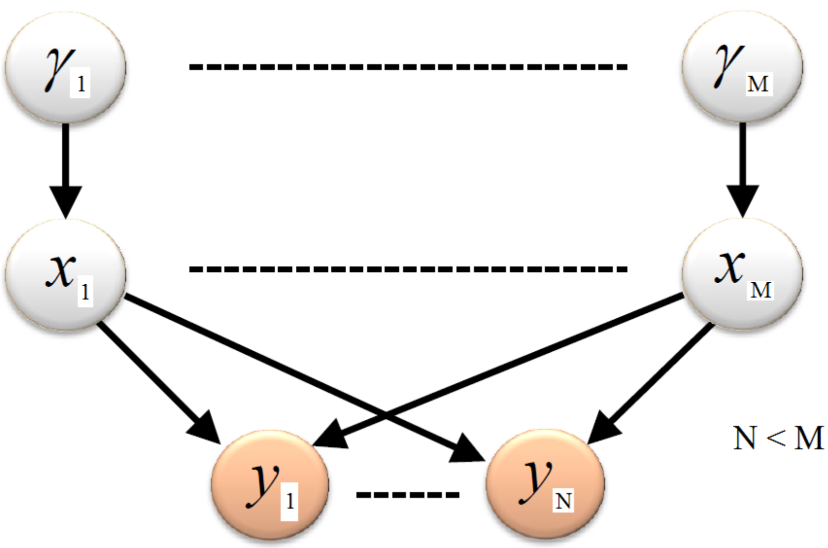


Problem Statement

$$\hat{x} = \arg\max_x P(x|y) = \arg\max_x P(y|x)P(x)$$

Choice of $P(x) = \frac{a}{2}e^{-a|x|}$ as Laplacian and $P(y|x)$ as Gaussian will lead to the familiar LASSO framework.

Hierarchical Bayesian Framework (Type II)



Hierarchical Bayesian Framework (Type II)

Problem Statement

$$\hat{\gamma} = \arg\max_{\gamma} P(\gamma|y) = \arg\max_{\gamma} \int P(y|x)P(x|\gamma)P(\gamma)dx$$

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

Hierarchical Bayesian Framework (Type II)

Problem Statement

$$\hat{\gamma} = \arg\max_{\gamma} P(\gamma|y) = \arg\max_{\gamma} \int P(y|x)P(x|\gamma)P(\gamma)dx$$

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

Example: Bayesian LASSO

Laplacian prior $P(x)$ can be represented as a Gaussian Scale Mixture in this fashion,

$$\begin{aligned} P(x) &= \int P(x|\gamma)P(\gamma)d\gamma \\ &= \int \frac{1}{\sqrt{2\pi\gamma}} \exp(-\frac{x^2}{2\gamma}) \times \frac{a^2}{2} \exp(-\frac{a^2}{2}\gamma)d\gamma \\ &= \frac{a}{2} exp(-a|x|) \end{aligned}$$

MAP Estimation

Problem Statement

$$\hat{x} = \arg\max_x P(x|y) = \arg\max_x P(y|x)P(x)$$

Advantages

- Many options to promote sparsity, i.e. choose some sparse prior over x.
- Growing options for solving the underlying optimization problem.
- Can be related to LASSO and other ℓ_1 minimization techniques by using suitable $P(x)$.

MAP Estimation

Assumption: Gaussian Noise

$$\begin{aligned}\hat{x} &= \arg\max_x P(y|x)P(x) \\ &= \arg\min_x -logP(y|x) - logP(x) \\ &= \arg\min_x \|y - \Phi x\|_2^2 + \lambda \sum_{i=1}^{m} g(|x_i|)\end{aligned}$$

MAP Estimation

Assumption: Gaussian Noise

$$
\begin{aligned}
\hat{x} &= \arg\max_{x} P(y|x)P(x) \\
&= \arg\min_{x} -logP(y|x) - logP(x) \\
&= \arg\min_{x} \|y - \Phi x\|_2^2 + \lambda \sum_{i=1}^{m} g(|x_i|)
\end{aligned}
$$

Theorem

If g is non decreasing and strictly concave function for $x \in \mathbb{R}^+$, the local minima of the above optimization problem will be the extreme points, i.e. have max of N non-zero entries.

Special cases of MAP estimation

Gaussian Prior

Gaussian assumption of $P(x)$ leads to ℓ_2 norm regularized problem

$$\hat{x} = \arg\min_{x} \|y - \Phi x\|_2^2 + \lambda \|x\|_2^2$$

Special cases of MAP estimation

Gaussian Prior

Gaussian assumption of $P(x)$ leads to ℓ_2 norm regularized problem

$$\hat{x} = \arg\min_x \|y - \Phi x\|_2^2 + \lambda \|x\|_2^2$$

Laplacian Prior

Laplacian assumption of $P(x)$ leads to standard ℓ_1 norm regularized problem i.e. LASSO.

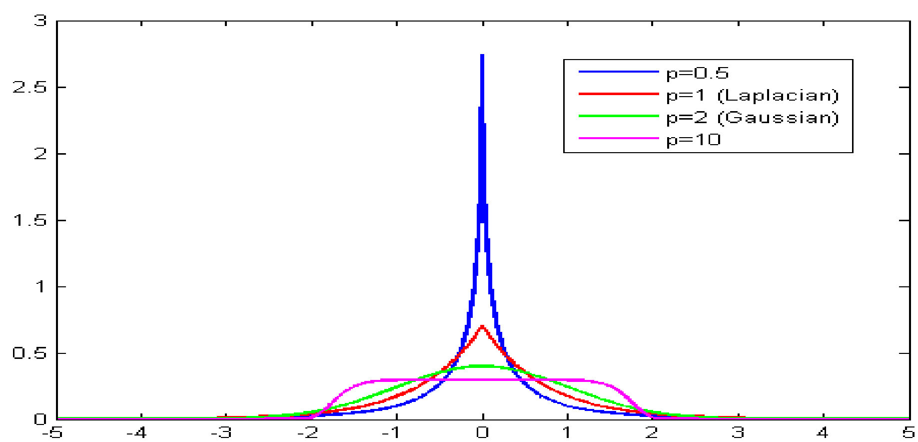
$$\hat{x} = \arg\min_x \|y - \Phi x\|_2^2 + \lambda \|x\|_1$$

Examples of Sparse Distributions

Sparse distributions can be viewed using a general framework of supergaussian distribution.

$$P(x;\beta,p) = \frac{p}{2\sqrt[p]{2}\beta\Gamma(\frac{1}{p})} e^{\frac{-|x|^p}{2\beta^p}}, \quad p \leq 1$$

If a unit variance distribution is desired β becomes a function of p.



Example of Sparsity Penalties

Practical Selections

$$\begin{aligned} g(x_i) &= \log(x_i^2+\epsilon), && \text{[Chartrand and Yin, 2008]} \\ g(x_i) &= \log(|x_i|+\epsilon), && \text{[Candes et al., 2008]} \\ g(x_i) &= |x_i|^p, && \text{[Rao et al., 1999]} \end{aligned}$$

Different choices favor different levels of sparsity.

Which Sparse prior to choose?

$$\hat{x} = \arg\min_{x} \|y - \Phi x\|_2^2 + \lambda \sum_{l=1}^{M} |x_l|^p$$

Two issues:

- If the prior is too sparse, i.e. $p \sim 0$, then we may get stuck at a local minima which results in convergence error.
- If the prior is not sparse enough, i.e. $p \sim 1$, then though global minima can be found, it may not be the sparsest solution, which results in a structural error.

MAP Estimation

Underlying Optimization problem is

$$\hat{x} = \arg\min_{x} \|y - \Phi x\|_2^2 + \lambda \sum_{i=1}^{m} g(|x_i|)$$

MAP Estimation

Underlying Optimization problem is

$$\hat{x} = \arg\min_{x} \|y - \Phi x\|_2^2 + \lambda \sum_{i=1}^{m} g(|x_i|)$$

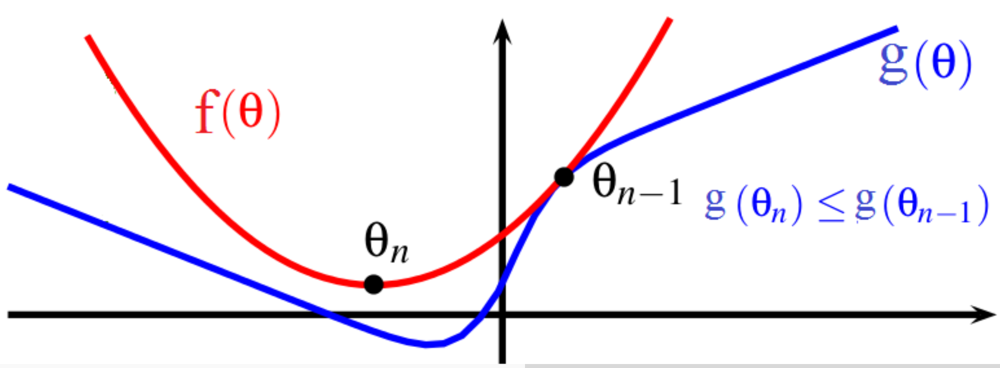
- Useful algorithms exist to minimize the cost function with a strictly concave penalty function g on R^+ (Reweighted ℓ_2/ℓ_1 algorithms).
- The essence of this algorithm is to create a bound for the concave penalty function and follow the steps of a Majorize-Minimization (MM) algorithm.

MAP Estimation

Underlying Optimization problem is

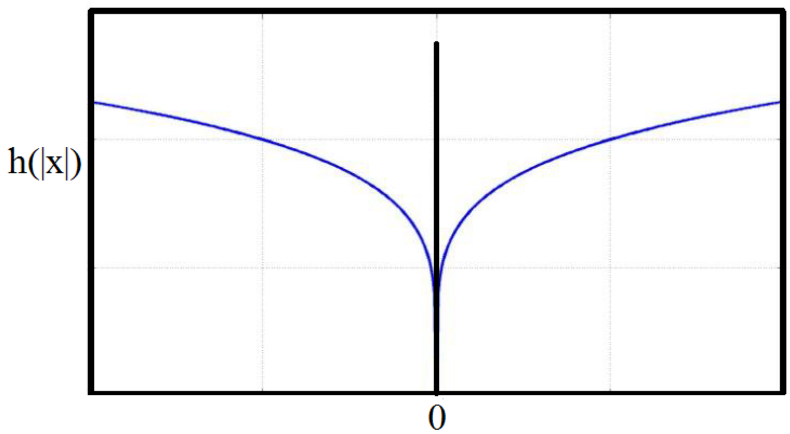
$$\hat{x} = \arg\min_x \|y - \Phi x\|_2^2 + \lambda \sum_{i=1}^{m} g(|x_i|)$$

- Useful algorithms exist to minimize the cost function with a strictly concave penalty function g on R^+ (Reweighted ℓ_2/ℓ_1 algorithms).
- The essence of this algorithm is to create a bound for the concave penalty function and follow the steps of a Majorize-Minimization (MM) algorithm.



Reweighted ℓ_1 optimization

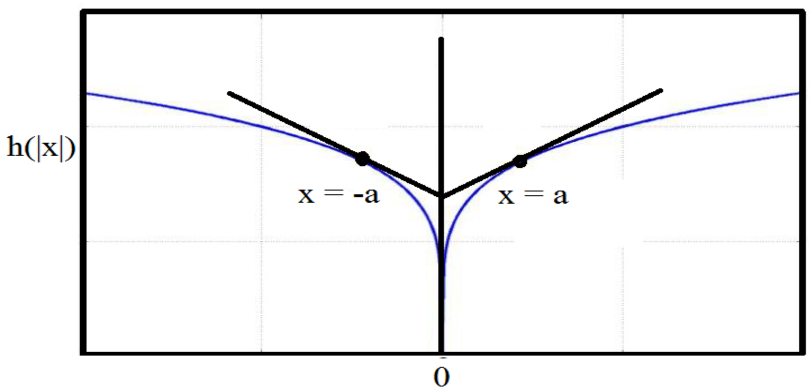
Assume: $g(x_i) = h(|x_i|)$ with h concave.



Now we have to bound this concave penalty function.

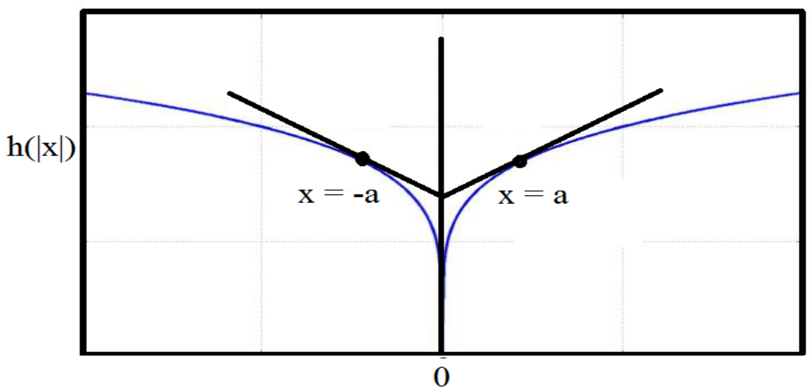
Reweighted ℓ_1 optimization

Assume: $g(x_i) = h(|x_i|)$ with h concave.



Reweighted ℓ_1 optimization

Assume: $g(x_i) = h(|x_i|)$ with h concave.



Updates

$$x^{(k+1)} \rightarrow argmin_x \|y - \Phi x\|_2^2 + \lambda \sum_i w_i^{(k)} |x_i|$$

$$w_i^{k+1} \rightarrow \frac{\partial g(x_i)}{\partial |x_i|} \mid_{x_i = x_i^{(k+1)}}$$

Reweighted ℓ_1 optimization

Candes et al., 2008

- Penalty: $g(x_i) = \log(|x_i| + \epsilon), \;\; 0\epsilon \geq 0$
- Weight Update: $w_i^{(k+1)} \rightarrow [|x_i^{(k+1)} + \epsilon]^{-1}$

Reweighted ℓ_2 optimization

- **Assume:** $g(x_i) = h(x_i^2)$ with h concave
- Upper bound $h(.)$ as before.
- Bound will be quadratic in the variables leading to a weighted 2-norm optimization problem

Reweighted ℓ_2 optimization

- **Assume:** $g(x_i) = h(x_i^2)$ with h concave
- Upper bound $h(.)$ as before.
- Bound will be quadratic in the variables leading to a weighted 2-norm optimization problem

Updates

$$x^{(k+1)} \rightarrow argmin_x \|y - \Phi x\|_2^2 + \lambda \sum_i w_i^{(k)} x_i^2$$
$$= \tilde{W}^{(k)} \Phi^T (\lambda I + \Phi \tilde{W}^{(k)} \Phi^T)^{-1} y$$

$$w_i^{k+1} \rightarrow \frac{\partial g(x_i)}{\partial x_i^2} |_{x_i = x_i^{(k+1)}} , \quad \tilde{W}^{(k+1)} \rightarrow diag[w^{(k+1)}]^{-1}$$

Reweighted ℓ_2 optimization: Examples

FOCUSS Algorithm[Rao et al., 2003]

- Penalty: $g(x_i) = |x_i|^p, \;\; 0 \leq p \leq 2$
- Weight Update: $w_i^{(k+1)} \rightarrow |x_i^{(k+1)}|^{p-2}$
- Properties: Well-characterized convergence rates; very susceptible to local minima when p is small.

Reweighted ℓ_2 optimization: Examples

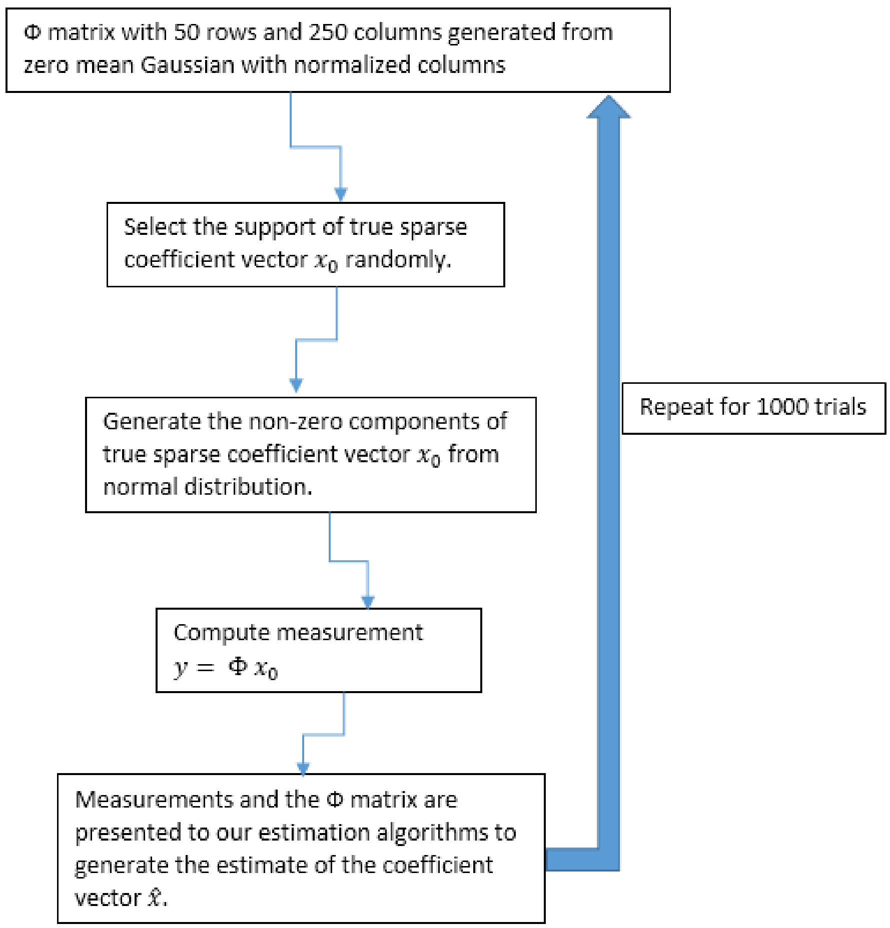
FOCUSS Algorithm[Rao et al., 2003]

- Penalty: $g(x_i) = |x_i|^p, \;\; 0 \leq p \leq 2$
- Weight Update: $w_i^{(k+1)} \rightarrow |x_i^{(k+1)}|^{p-2}$
- Properties: Well-characterized convergence rates; very susceptible to local minima when p is small.

Chartrand and Yin (2008) Algorithm

- Penalty: $g(x_i) = \log(x_i^2 + \epsilon), \;\; \epsilon \geq 0$
- Weight Update: $w_i^{(k+1)} \rightarrow [(x_i^{(k+1)})^2 + \epsilon]^{-1}$
- Properties: Slowly reducing ϵ to zero smoothes out local minima initially allowing better solutions to be found;

Empirical Comparison



Empirical Comparison

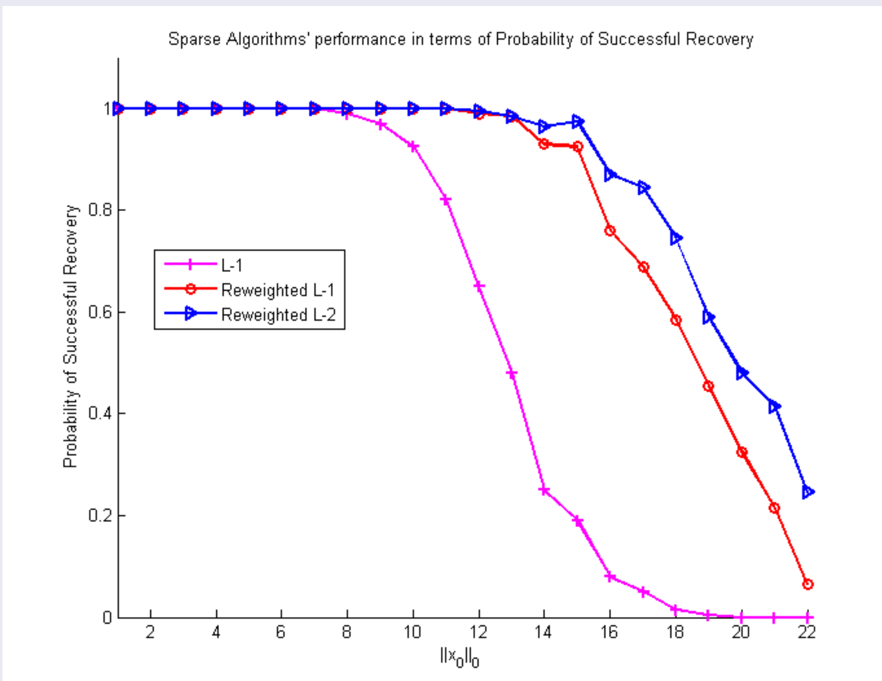


Figure: Probability of Successful recovery vs Number of non zero coefficients

Limitation of MAP based methods

To retain the same maximally sparse global solution as the ℓ_0 norm in general conditions, then any possible MAP algorithm will possess $O\left[\binom{M}{N}\right]$ local minima.

Hierarchical Bayes: Sparse Bayesian Learning(SBL)

Hierarchical Bayes: Sparse Bayesian Learning(SBL)

MAP estimation is just a penalized regression, hence Bayesian Interpretation has not contributed much as of now.

Hierarchical Bayes: Sparse Bayesian Learning(SBL)

MAP estimation is just a penalized regression, hence Bayesian Interpretation has not contributed much as of now.

MAP methods were interested in the mode of the posterior but SBL uses posterior information beyond the mode, i.e. posterior distribution.

Hierarchical Bayes: Sparse Bayesian Learning(SBL)

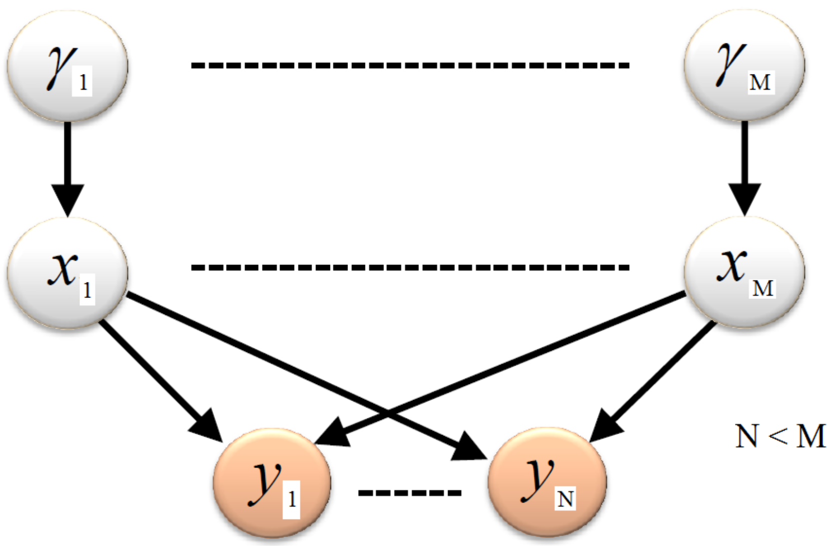
MAP estimation is just a penalized regression, hence Bayesian Interpretation has not contributed much as of now.

MAP methods were interested in the mode of the posterior but SBL uses posterior information beyond the mode, i.e. posterior distribution.

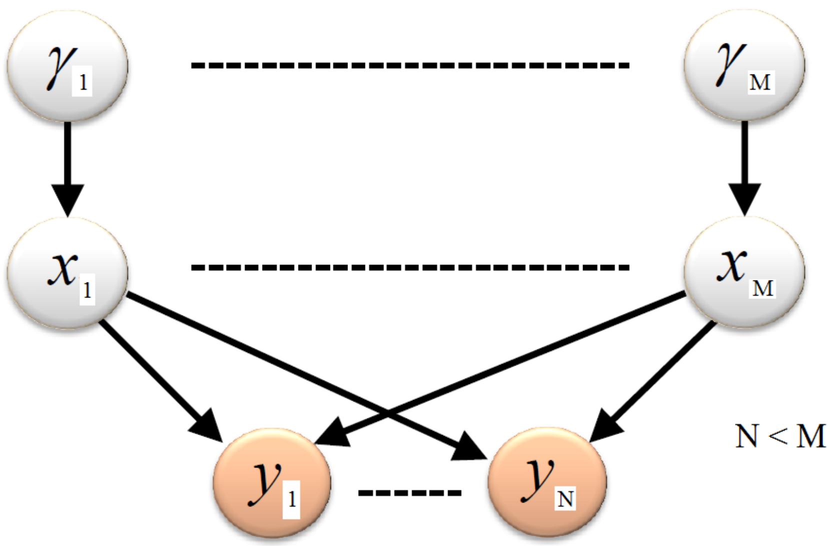
Problem

For all sparse priors it is not possible to compute the normalized posterior $P(x|y)$, hence some approximations are needed.

Hierarchical Bayesian Framework (Type II)



Hierarchical Bayesian Framework (Type II)



In order for this framework to be useful, we need tractable representations: Gaussian Scaled Mixtures

Construction of Sparse priors

Separability: $P(x) = \prod_i P(x_i)$

Construction of Sparse priors

Separability: $P(x) = \prod_i P(x_i)$

Gaussian Scale Mixture :

$$P(x_i) = \int P(x_i|\gamma_i)P(\gamma_i)d\gamma_i = \int N(x_i; 0, \gamma_i)P(\gamma_i)d\gamma_i$$

Construction of Sparse priors

Separability: $P(x) = \prod_i P(x_i)$

Gaussian Scale Mixture :

$$P(x_i) = \int P(x_i|\gamma_i)P(\gamma_i)d\gamma_i = \int N(x_i; 0, \gamma_i)P(\gamma_i)d\gamma_i$$

Most of the sparse priors over x (including those with concave g) can be represented in this GSM form, and different scale mixing density i.e, $P(\gamma_i)$ will lead to different sparse priors. [Palmer et al., 2006]

Construction of Sparse priors

Separability: $P(x) = \prod_i P(x_i)$

Gaussian Scale Mixture :

$$P(x_i) = \int P(x_i|\gamma_i)P(\gamma_i)d\gamma_i = \int N(x_i; 0, \gamma_i)P(\gamma_i)d\gamma_i$$

Most of the sparse priors over x (including those with concave g) can be represented in this GSM form, and different scale mixing density i.e, $P(\gamma_i)$ will lead to different sparse priors. [Palmer et al., 2006]

Instead of solving a MAP problem in x, in the Bayesian framework one estimates the hyperparameters γ leading to an estimate of the posterior distribution for x, i.e. $P(x|y; \hat{\gamma})$. (Sparse Bayesian Learning)

Examples of Gaussian Scale Mixture

Laplacian density

$$P(x;a) = \frac{a}{2} exp(-a|x|)$$

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

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.

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 $p/2$.

Sparse Bayesian Learning (Tipping)

$$y = \Phi x + v$$

Solving for the optimal γ

$$\hat{\gamma} = \arg\max_{\gamma} P(\gamma|y) = \arg\max_{\gamma} P(y|\gamma)P(\gamma)$$
$$= \arg\min_{\gamma} log|\Sigma_y| + y^T\Sigma_y^{-1}y - 2\sum_i logP(\gamma_i)$$

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

Sparse Bayesian Learning (Tipping)

$$y = \Phi x + v$$

Solving for the optimal γ

$$\hat{\gamma} = \arg\max_{\gamma} P(\gamma|y) = \arg\max_{\gamma} P(y|\gamma)P(\gamma)$$
$$= \arg\min_{\gamma} log|\Sigma_y| + y^T\Sigma_y^{-1}y - 2\sum_i logP(\gamma_i)$$

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

Empirical Bayes

Choose $P(\gamma_i)$ to be a non-informative prior

Sparse Bayesian Learning

Computing Posterior

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

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

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

Sparse Bayesian Learning

Computing Posterior

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

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

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

Updating γ

Using EM algorithm with a non informative prior over γ, the update rule becomes:

$$\gamma_i \leftarrow \mu_x(i)^2 + \Sigma_x(i,i)$$

SBL properties

SBL properties

- Local minima are sparse, i.e. have at most N nonzero γ_i

SBL properties

- Local minima are sparse, i.e. have at most N nonzero γ_i
- Bayesian inference cost is generally much smoother than associated MAP estimation. Fewer local minima.

SBL properties

- Local minima are sparse, i.e. have at most N nonzero γ_i
- Bayesian inference cost is generally much smoother than associated MAP estimation. Fewer local minima.
- In high signal to noise ratio, the global minima is the sparsest solution. No structural problems.

Empirical Comparison

For each test case

1. Generate a random dictionary Φ with 50 rows and 250 columns from the normal distribution and normalize each column to have 2-norm of 1.
2. Select the support for the true sparse coefficient vector x_0 randomly.
3. Generate the non-zero components of x_0 from the normal distribution.
4. Compute signal, $y = \Phi x_0$ (Noiseless case).
5. Compare SBL with previous methods with regard to estimating x_0.
6. Average over 1000 independent trials.

Empirical Comparison: 1000 trials

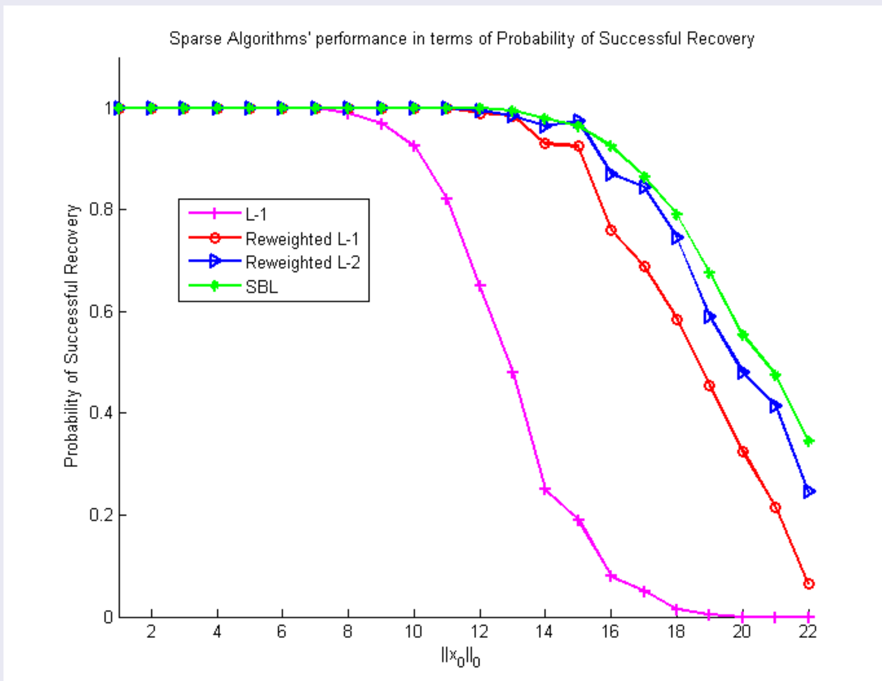


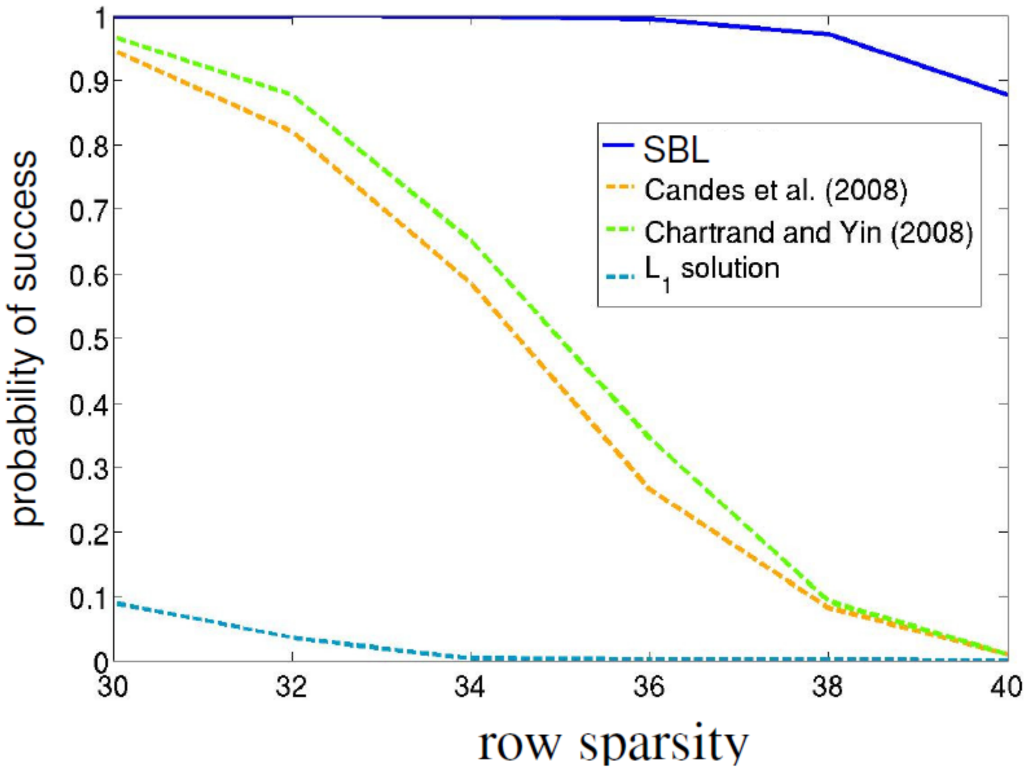
Figure: Probability of Successful recovery vs Number of non zero coefficients

Empirical Comparison: Multiple Measurement Vectors (MMV)

Generate data matrix via $Y = \Phi X_0$ (noiseless), where:

1. X_0 is 100-by-5 with random non-zero rows.
2. Φ is 50-by-100 with Gaussian iid entries.

Empirical Comparison: 1000 trials



Summary

Summary

- Bayesian methods offer interesting and useful options to the Sparse Signal Recovery problem
  - MAP estimation (Reweighted ℓ_2/ℓ_1 algorithms)
  - Sparse Bayesian Learning

Summary

- Bayesian methods offer interesting and useful options to the Sparse Signal Recovery problem
  - MAP estimation (Reweighted ℓ_2/ℓ_1 algorithms)
  - Sparse Bayesian Learning
- Versatile and can be more easily employed in problems with structure

Summary

- Bayesian methods offer interesting and useful options to the Sparse Signal Recovery problem
  - MAP estimation (Reweighted ℓ_2/ℓ_1 algorithms)
  - 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.