Bayesian Methods for Sparse Signal Recovery

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

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

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.

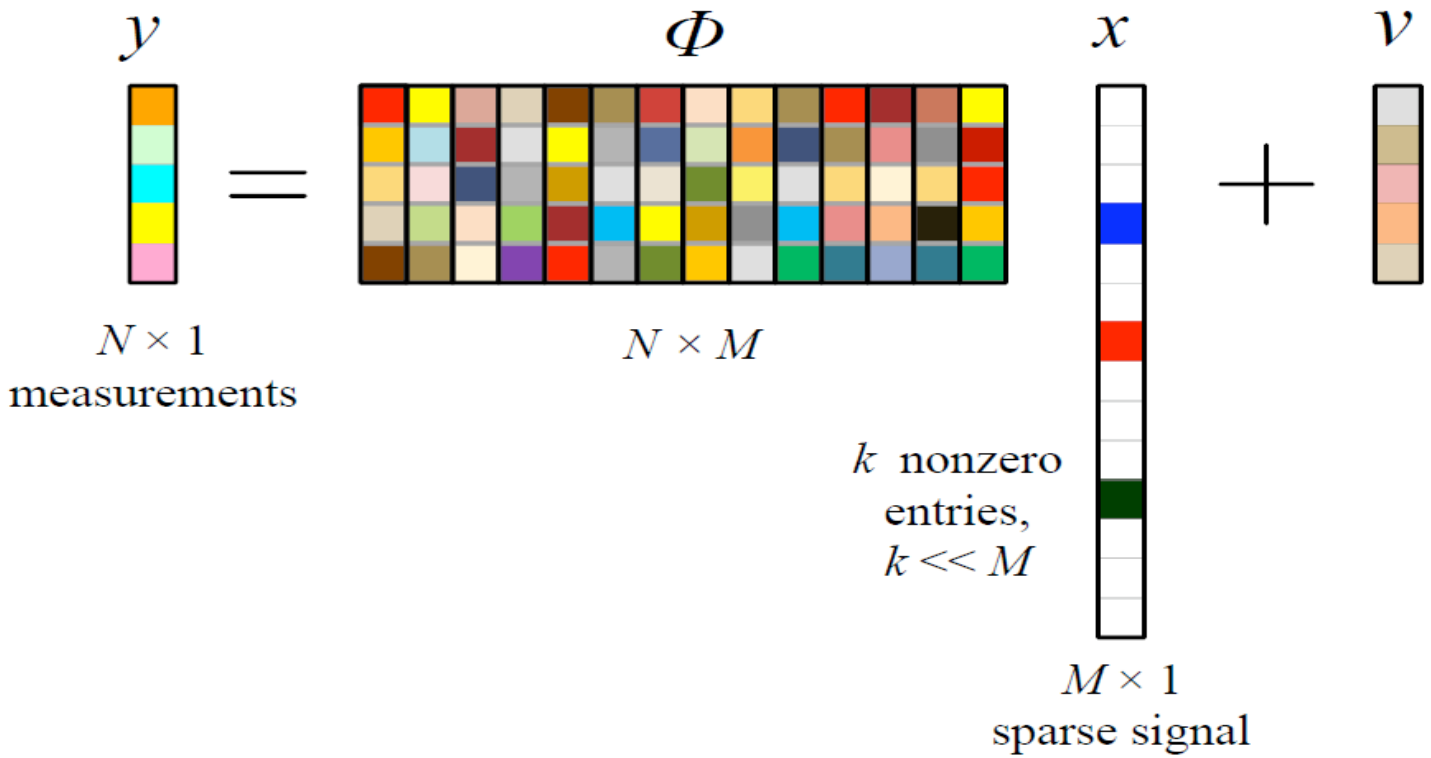
Many interesting developments in recent past that make the subject timely.

Bayesian Framework offers some interesting options.

Outline

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

Problem Description: Sparse Signal Recovery (SSR)



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

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

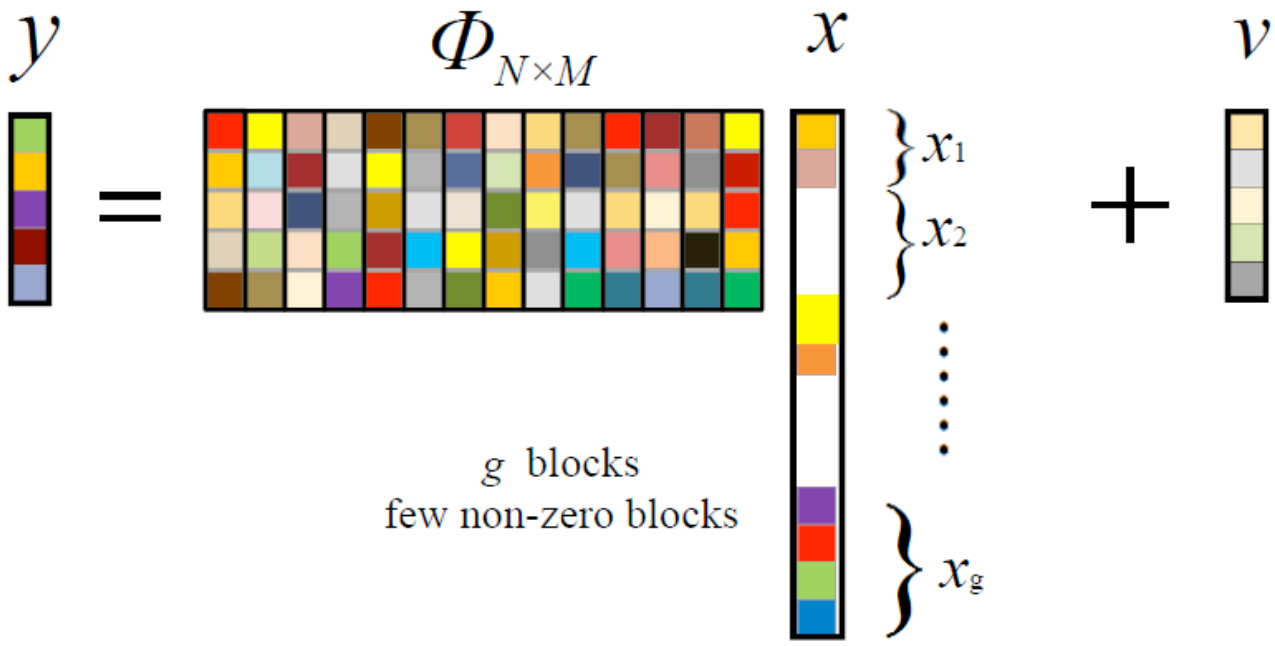
Given a target signal y and dictionary Φ, find the weights x that solve,

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

Useful Extensions

1. Block Sparsity
2. Multiple Measurement Vectors (MMV)
3. Block MMV
4. MMV with time varying sparsity

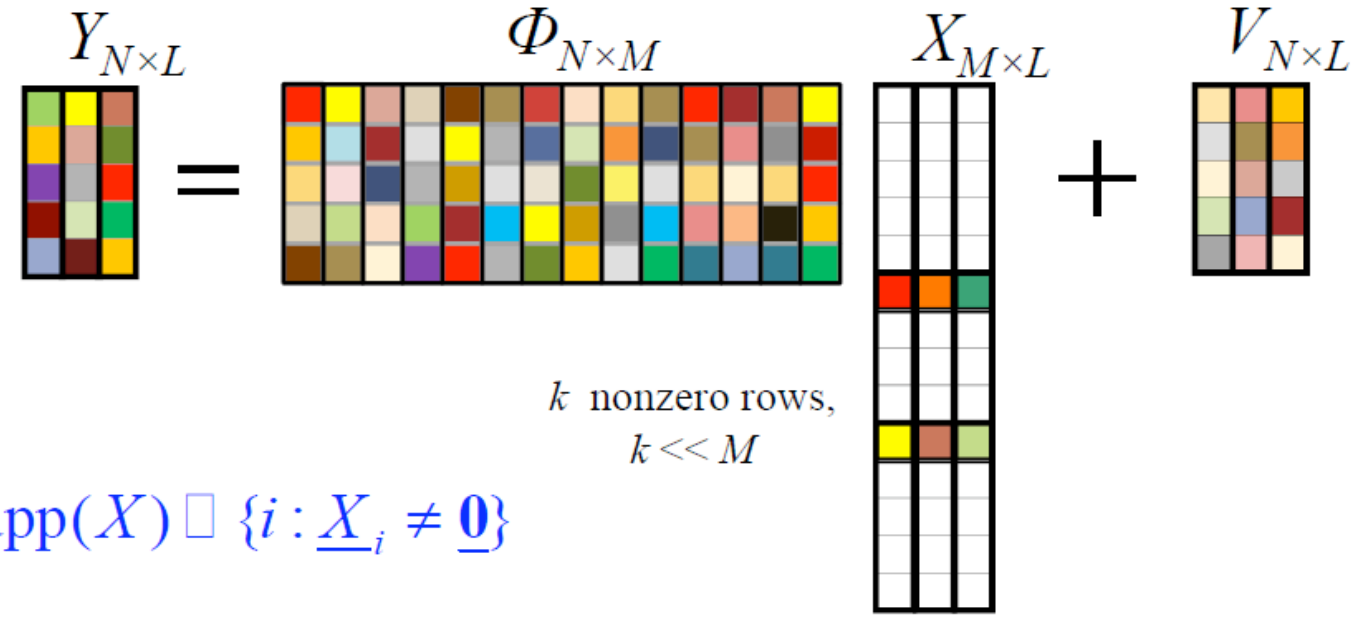
Block Sparsity



Support Recovery Problem: Given y and Φ, recover $\text{supp}(x)$

Multiple Measurement Vectors (MMV)

- Model



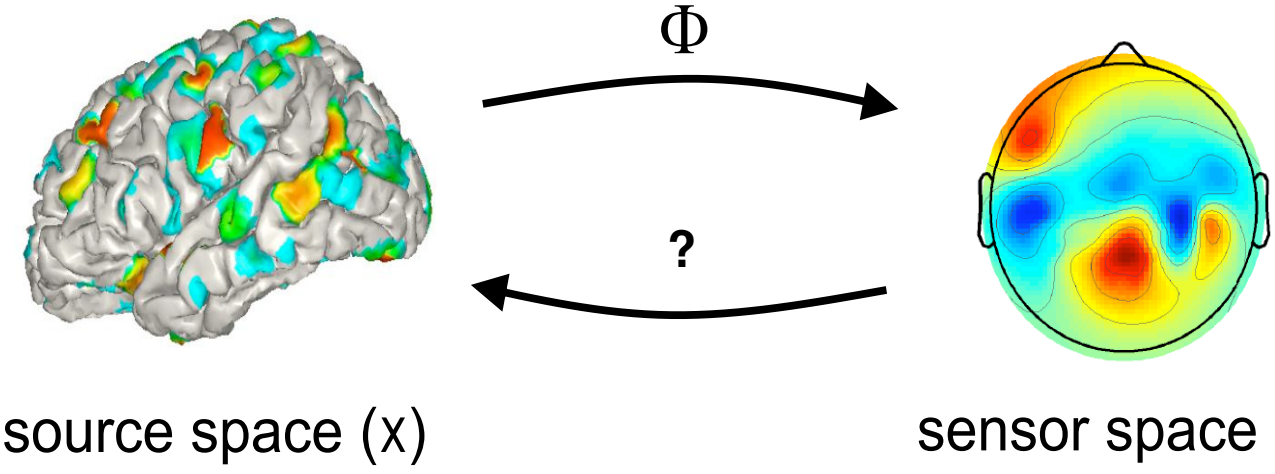
  - Multiple measurements: L measurements
  - Common Sparsity Profile: k nonzero rows

Applications

1. Signal Representation (Mallat, Coifman, Donoho,..)
2. EEG/MEG (Leahy, Gorodnitsky,Loannides,..)
3. Robust Linear Regression and Outlier Detection
4. Speech Coding (Ozawa, Ono, Kroon,..)
5. Compressed Sensing (Donoho, Candes, Tao,..)
6. Magnetic Resonance Imaging (Lustig,..)
7. Sparse Channel Equalization (Fevrier, Proakis,...)

and many more.....

MEG/EEG Source Localization

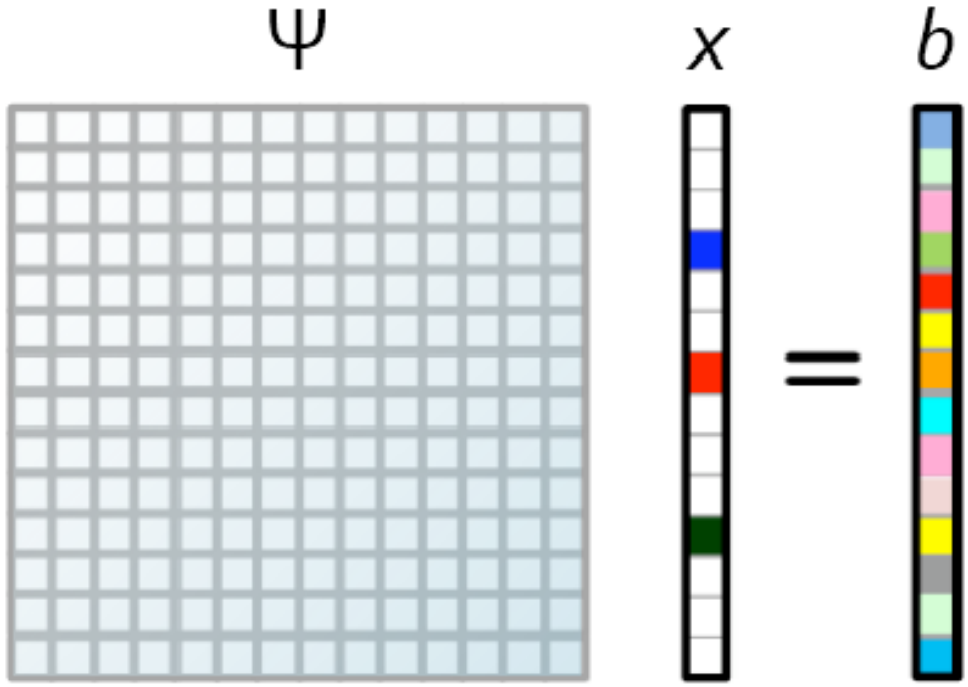


- Forward model dictionary Φ can be computed using Maxwell's equations [Sarvas,1987].
- In many situations the active brain regions may be relatively sparse, and so solving a sparse inverse problem is required.

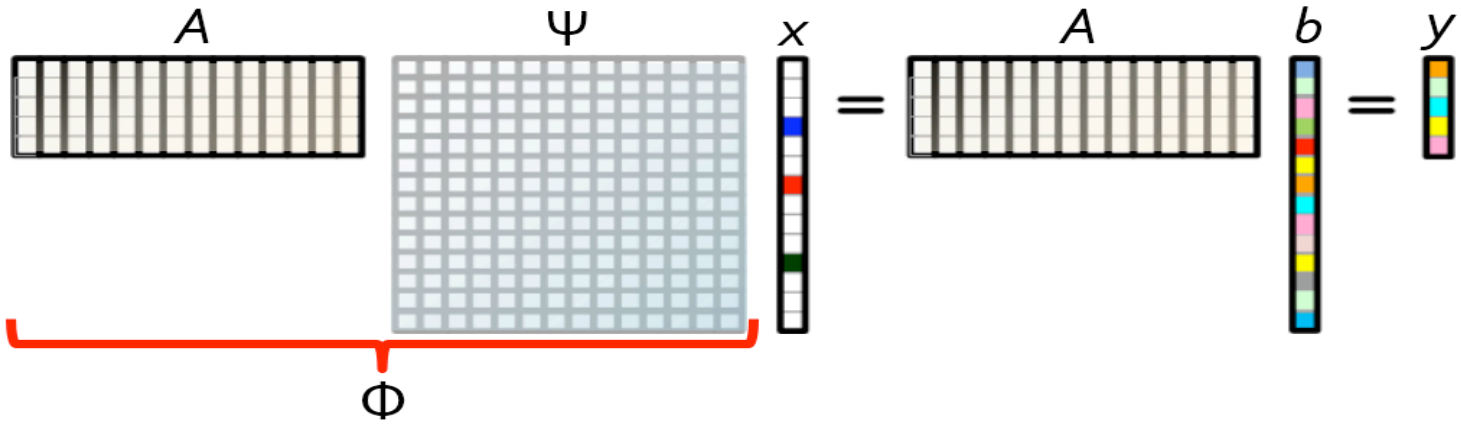
[Baillet et al., 2001]

Compressive Sampling (CS)

Transform Coding



Compressive Sampling (CS)



Computation:

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

Issues:

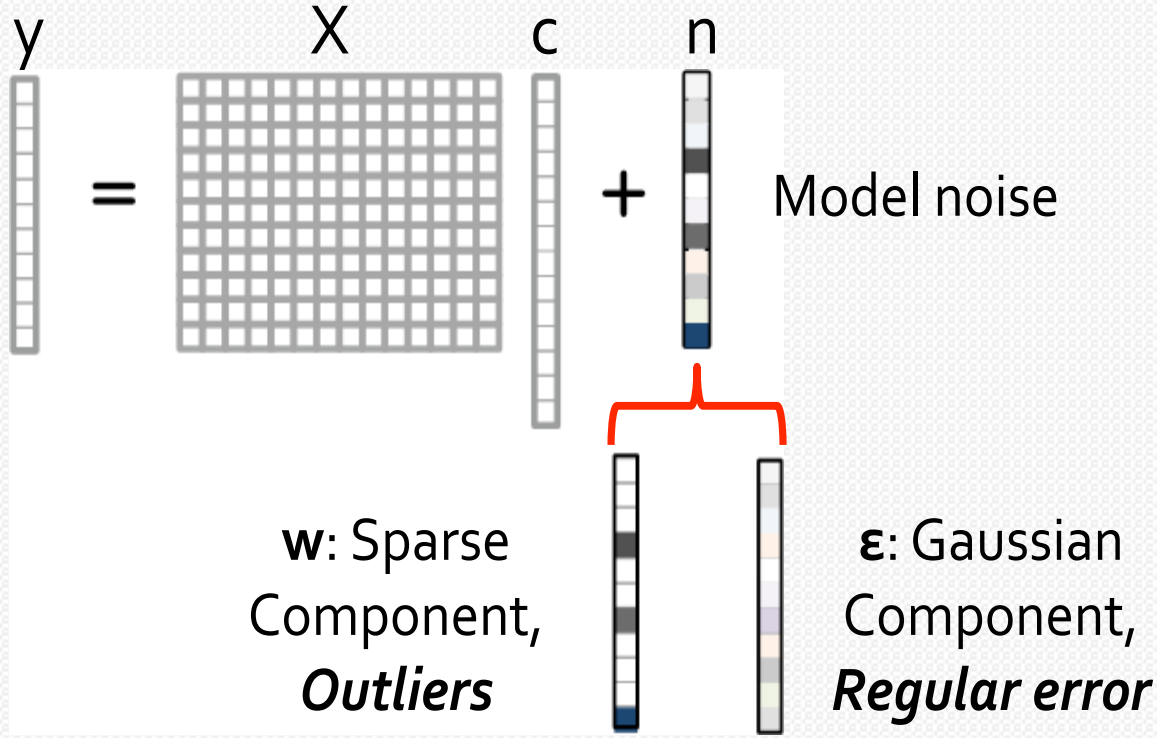
1. Need to recover sparse signal x with constraint $\Phi x = y$.
2. Need to design sampling matrix A.

Robust Linear Regression

X, y: data;

c: regression coeffs.;

n: model noise;



Transform into overcomplete representation:

$$\mathbf{Y = X\,c + \Phi\,w + \varepsilon}, \text{ where } \mathbf{\Phi} = I,$$

or

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

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

Problem Statement

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

Advantages

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

MAP Estimation

Assumption: Gaussian Noise

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

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

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

Laplacian Prior

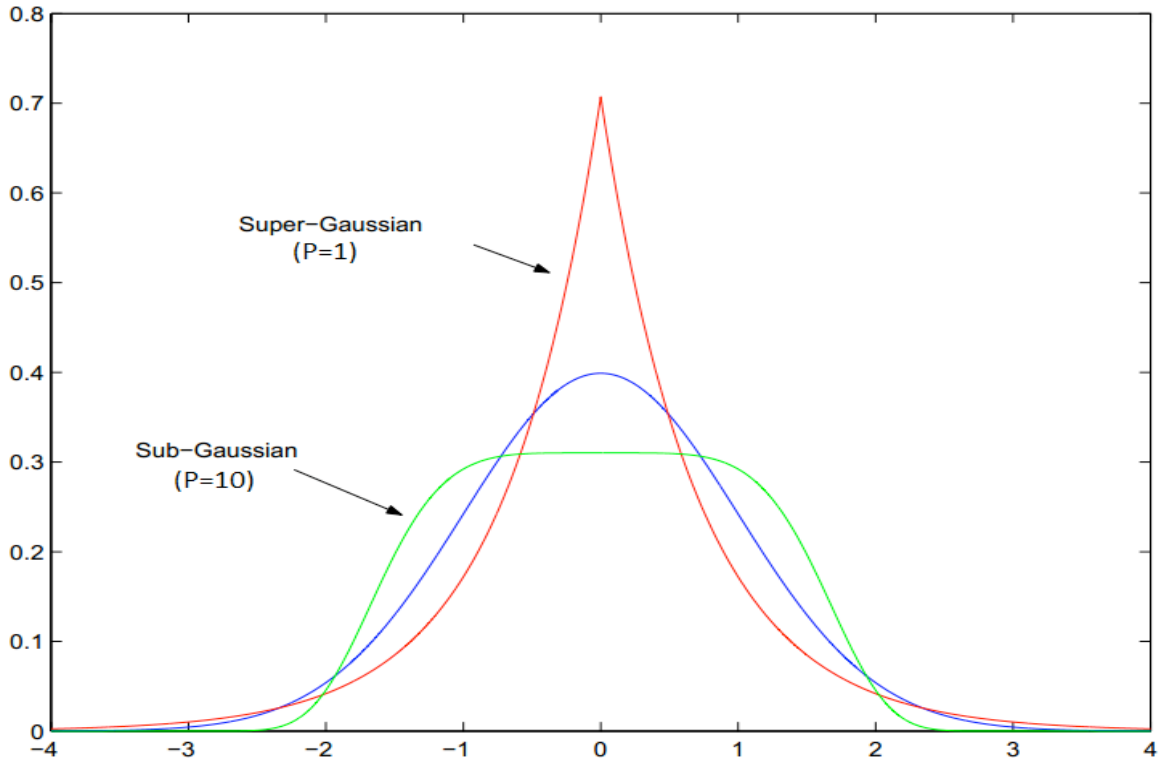
Laplacian assumption of $P(x)$ leads to standard ℓ_1 norm regularized problem i.e. LASSO.

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

Examples of Sparse Distributions

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

$$P(x) \propto e^{-|x|^p}, \quad p \leq 1$$



Example of Sparsity Penalties

Practical Selections

$$
\begin{array}{lcll}
g(x_i) & = & \log(x_i^2 + \epsilon), & \text{[Chartrand and Yin, 2008]} \\
g(x_i) & = & \log(|x_i| + \epsilon), & \text{[Candes et al., 2008]} \\
g(x_i) & = & |x_i|^p, & \text{[Rao et al., 2003]}
\end{array}
$$

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:

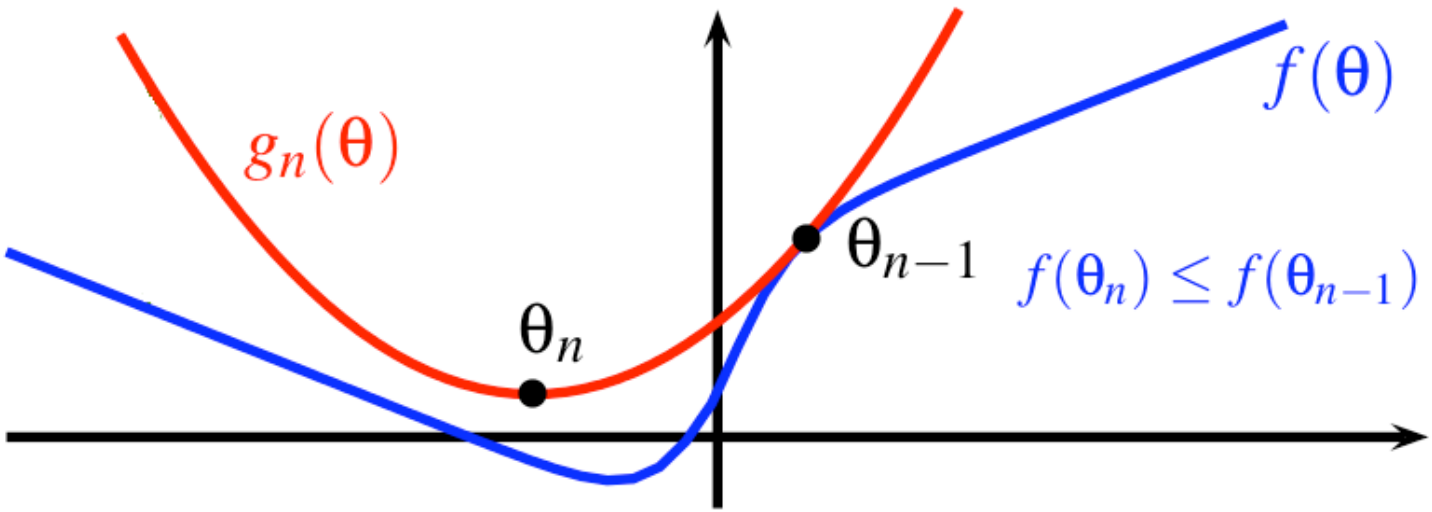
1. If the prior is too sparse, i.e. $p \sim 0$, then we may get stuck at a local minima which results in convergence error.
2. If the prior is not sparse enough, i.e. $p \sim 1$, then though global minima can be found, it may not be the sparsest solution, which results in a structural error.

Reweighted ℓ_2/ℓ_1 optimization

Underlying Optimization problem is

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

1. Useful algorithms exist to minimize the original cost function with a strictly concave penalty function g on R^+.
2. 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_2 optimization

Assume: $g(x_i) = h(x_i^2)$ with h concave.

Updates

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

$$= \tilde{W}^{(k)} \Phi^T (\lambda I + \Phi \tilde{W}^{(k)} \Phi^T)^{-1} y$$

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

Reweighted ℓ_2 optimization: Examples

FOCUSS Algorithm[Rao et al., 2003]

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

Chartrand and Yin (2008) Algorithm

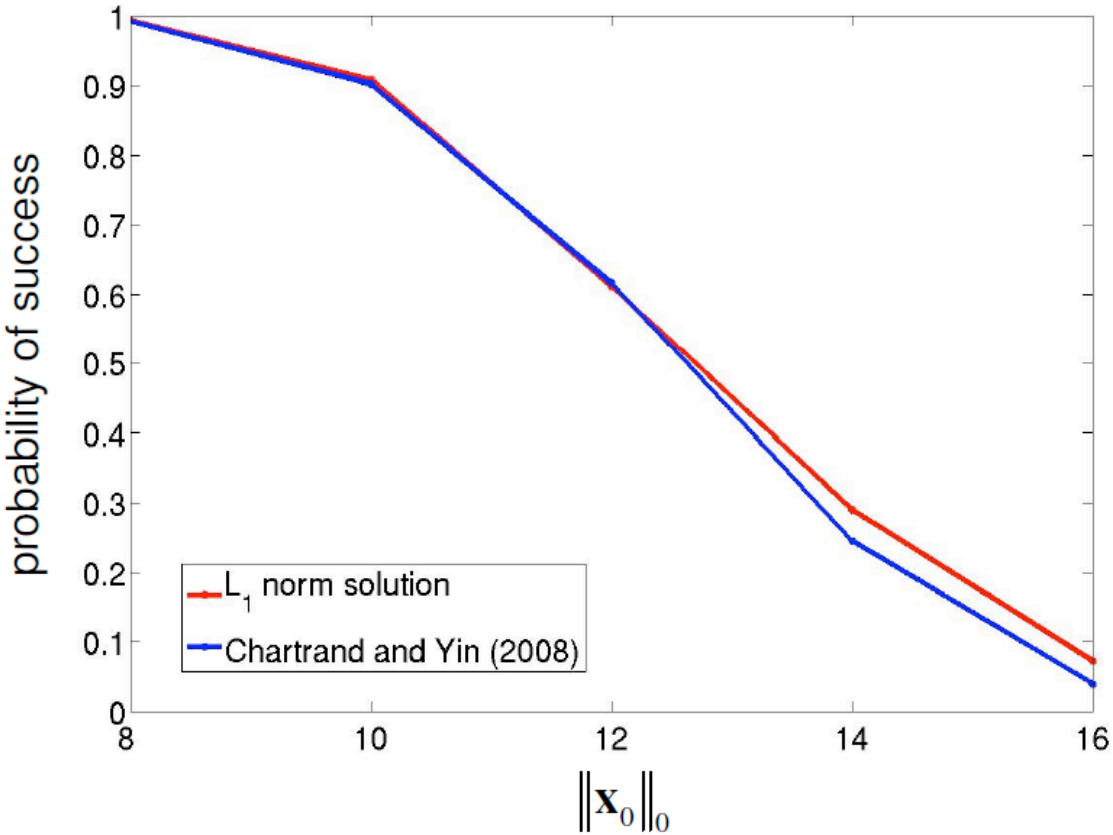
1. Penalty: $g(x_i) = \log(x_i^2 + \epsilon), \;\; \epsilon \geq 0$
2. Weight Update: $w_i^{(k+1)} \rightarrow [(x_i^{(k+1)})^2 + \epsilon]^{-1}$
3. Properties: Slowly reducing ϵ to zero smoothes out local minima initially allowing better solutions to be found;

Empirical Comparison

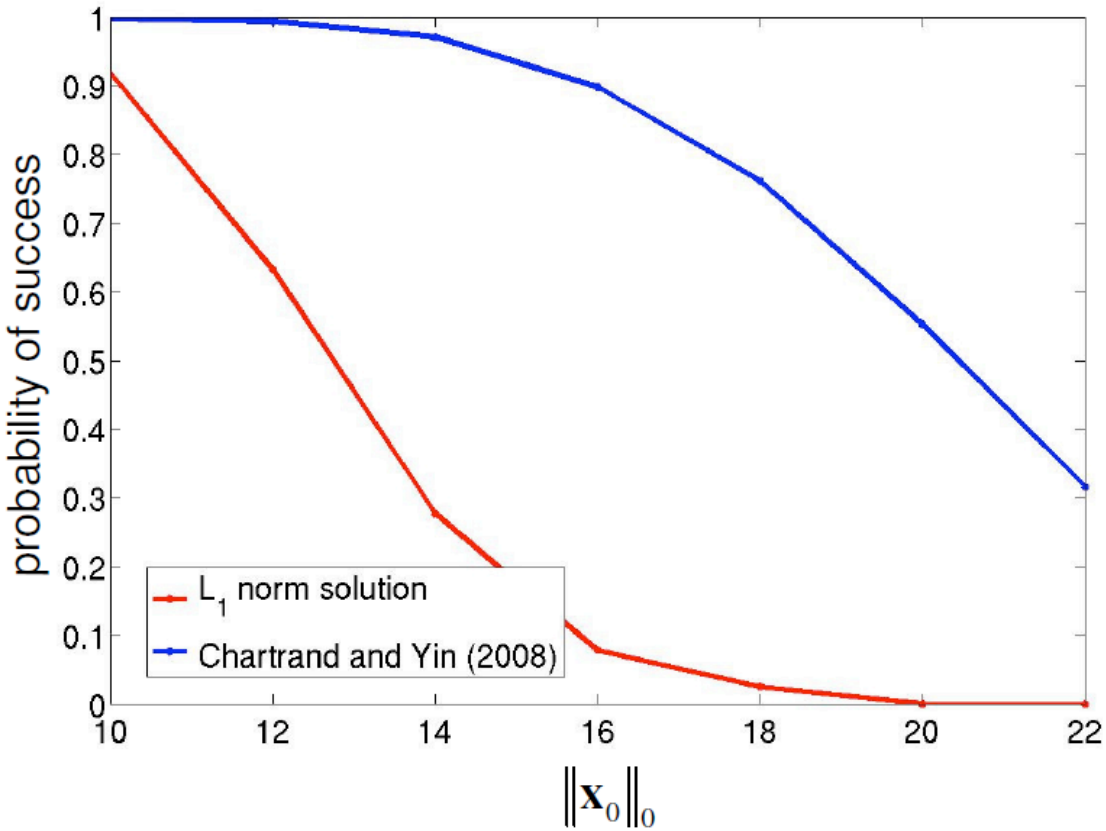
For each test case

1. Generate a random dictionary Φ with 50 rows and 250 columns.
2. Generate a sparse coefficient vector x_0.
3. Compute signal, $y = \Phi x_0$ (Noiseless case).
4. Compare Chartrand and Yin's reweighted ℓ_2 method with ℓ_1 norm solution with regard to estimating x_0.
5. Average over 1000 independent trials.

Empirical Comparison: Unit nonzeros



Empirical Comparison: Gaussian nonzeros



Reweighted ℓ_1 optimization

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

Updates

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

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

Reweighted ℓ_1 optimization

Candes et al., 2008

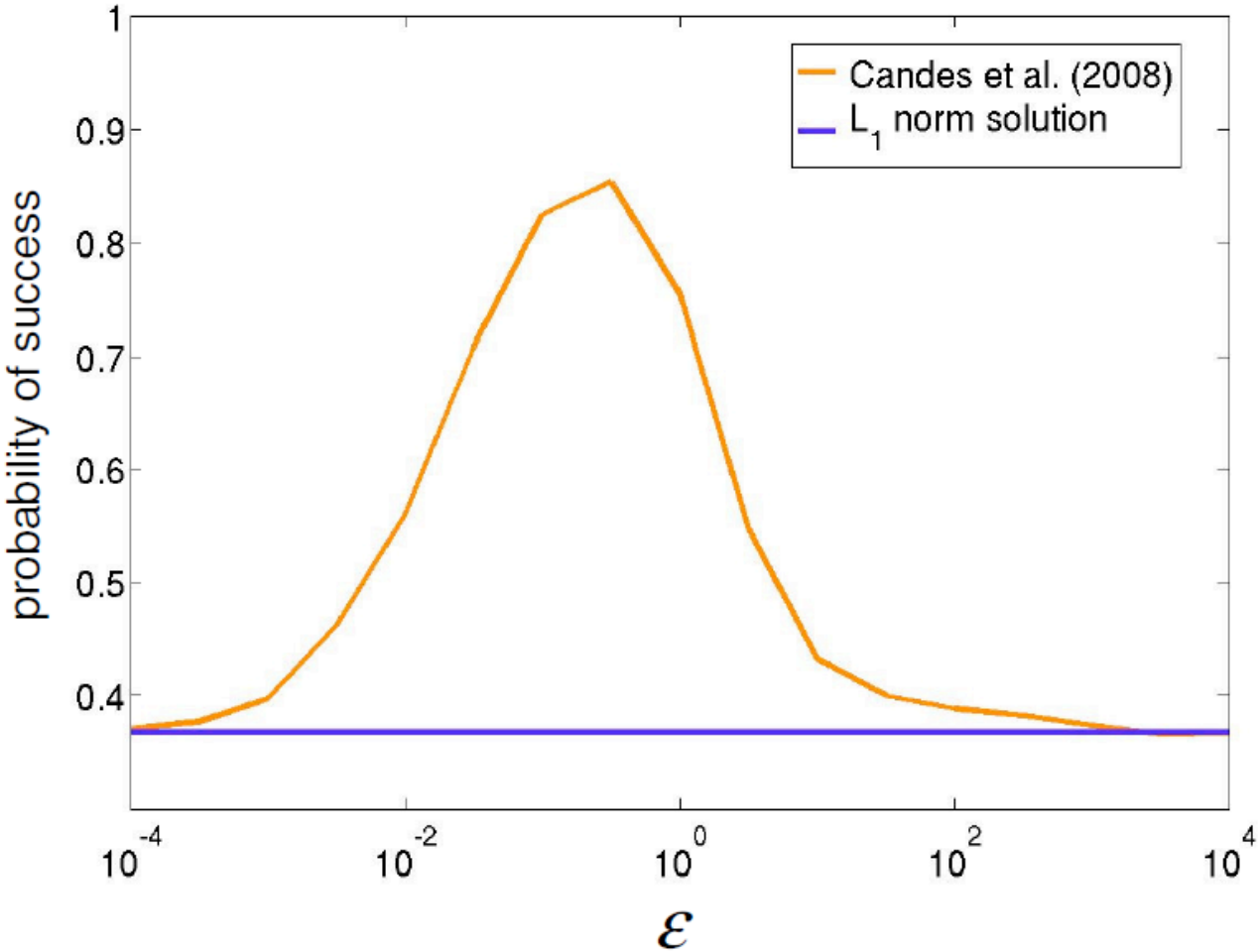
1. Penalty: $g(x_i) = \log(|x_i| + \epsilon), \quad 0\epsilon \geq 0$
2. Weight Update: $w_i^{(k+1)} \rightarrow [|x_i^{(k+1)} + \epsilon]^{-1}$

Empirical Comparison

For each test case

1. Generate a random dictionary Φ with 50 rows and 100 columns.
2. Generate a sparse coefficient vector x_0 with 30 truncated Gaussian, strictly positive nonzero coefficients.
3. Compute signal, $y = \Phi x_0$ (Noiseless case).
4. Compare Candes et al's reweighted ℓ_1 method with ℓ_1 norm solution, both constrained to be non-negative with regard to estimating x_0.
5. Average over 1000 independent trials.

Empirical Comparison



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.

Bayesian Inference: Sparse Bayesian Learning(SBL)

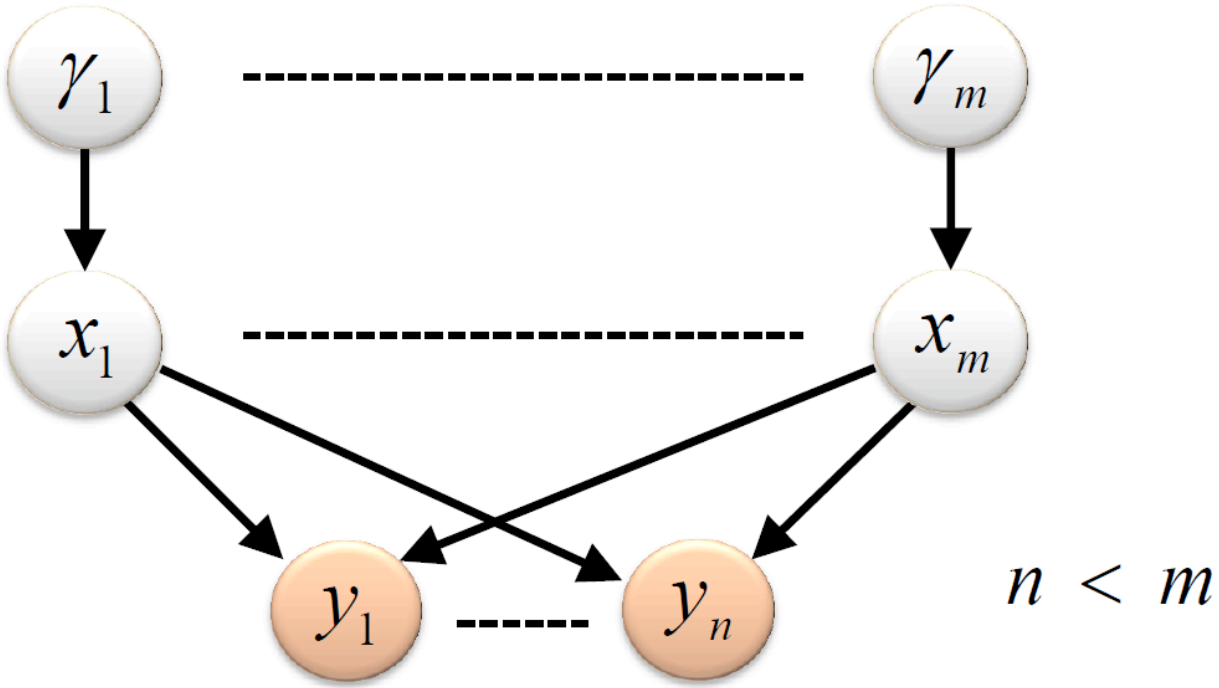
MAP estimation is just a penalized regression, hence Bayesian Interpretation has not contributed much as of now.

Previous 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 Bayes



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. (Sparse Bayesian Learning)

Examples of Gaussian Scale Mixture

Generalized Gaussian

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

Scale mixing density: Positive alpha stable density of order $\rho/2$.

Generalized Cauchy

$$p(x;\alpha,\nu) = \frac{\alpha\Gamma(\nu+1/\alpha)}{2\Gamma(1/\alpha)\Gamma(\nu)} \frac{1}{(1+|x|^{\alpha})^{\nu+1/\alpha}}$$

Scale mixing density: Gamma Distribution.

Examples of Gaussian Scale Mixture

Generalized Logistic

$$p(x;\alpha) = \frac{\Gamma(2\alpha)}{\Gamma(\alpha)^2} \frac{e^{-\alpha x}}{(1+e^{-x})^{2\alpha}}$$

Scale mixing density: Related to Kolmogorov-Smirnov distance statistic.

Sparse Bayesian Learning

$$y = \Phi x + v$$

Solving for the optimal γ

$$\begin{aligned}\hat{\gamma} &= \arg\max_{\gamma} P(\gamma|y) = \arg\max_{\gamma} \int P(y|x)P(x|\gamma)P(\gamma)dx \\ &= \arg\min_{\gamma} log|\Sigma_y| + y^T\Sigma_y^{-1}y - 2\sum_i logP(\gamma_i)\end{aligned}$$

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,

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

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

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

- 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.

Connection to MAP formulation

Using the relationship,

$$y\Sigma_y^{-1}y = \min_x \frac{1}{\lambda}\|y - \Phi x\|^2 + x^T\Gamma^{-1}x$$

x-space cost function becomes,

$$L_{II}^x(x) = \|y - \Phi x\|_2^2 + \lambda g_{II}(x)$$

where,

$$g_{II}(x) = \min_\gamma \sum_i \frac{x_i^2}{\gamma_i} + log|\Sigma_y| + \sum_i f(\gamma_i)$$

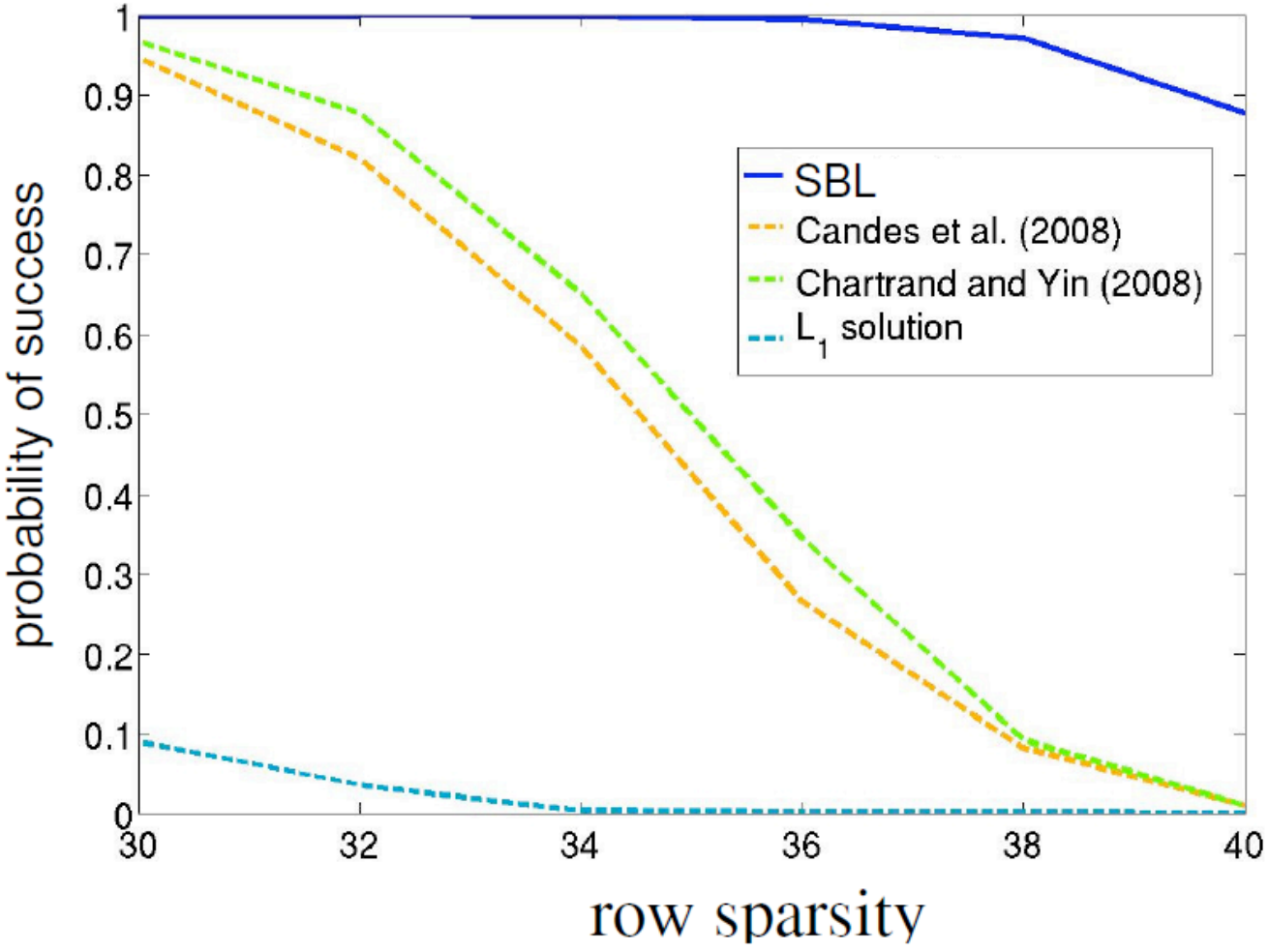
with, $f(\gamma_i) = -2logP(\gamma_i)$

Empirical Comparison: Simultaneous Sparse Approximation

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.

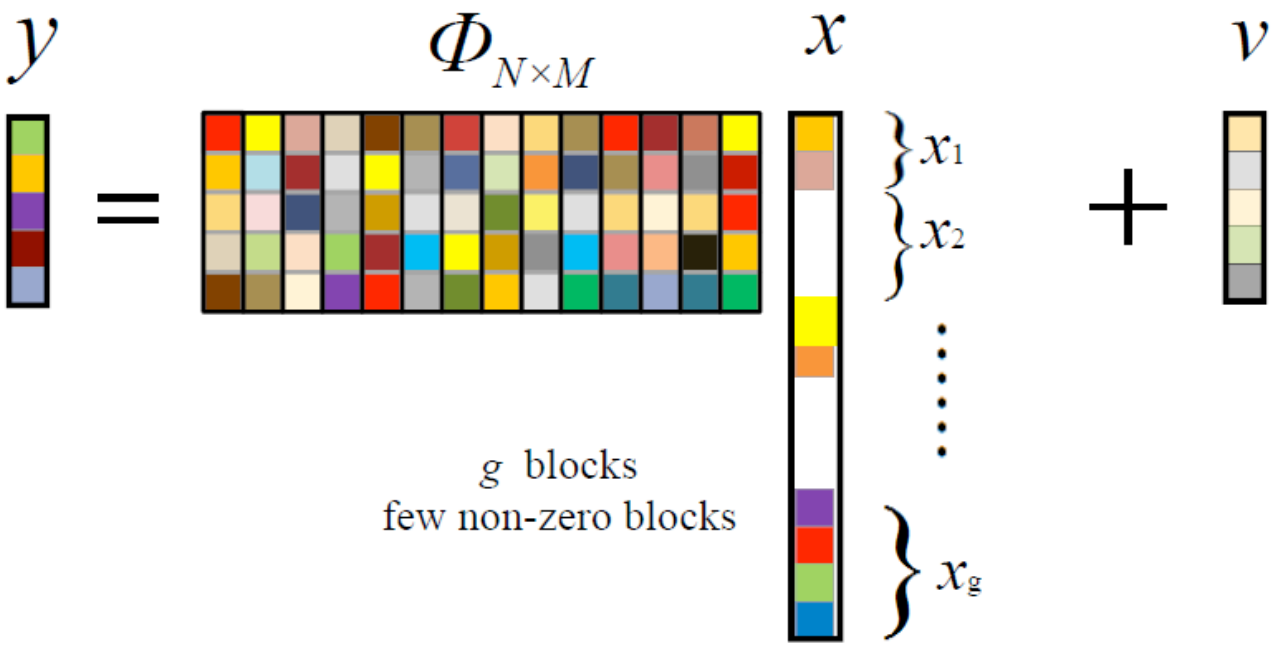
Empirical Comparison: 1000 trials



Useful Extensions

1. Block Sparsity
2. Multiple Measurement Vectors (MMV)
3. Block MMV
4. MMV with time varying sparsity

Block Sparsity



Support Recovery Problem: Given y and Φ, recover $\text{supp}(x)$

Intra-Vector Correlation is often present and is hard to model.

Block-Sparse Bayesian Learning Framework

Model

$$y = \Phi x + v$$

$$x = [\underbrace{x_1, ..., x_{d_1}}_{x_1^T} ....., \underbrace{x_{d_{g-1}+1}, ..., x_{d_g}}_{x_g^T}]^T$$

Parameterized Prior

$$P(x_i; \gamma_i, B_i) \sim N(0, \gamma_i B_i), \quad \text{where, } i = 1, ..., g$$

$$P(x; (\gamma_i, B_i)_i) \sim N(0, \Sigma_0)$$

γ_i: Control Block-Sparsity;
B_i: Capture intra-block correlation;

BSBL framework

Noise Model

$$P(v;\lambda) \sim N(0, \lambda I)$$

Posterior

$$P(x|y;\lambda,(\gamma_i, B_i)_{i=1}^{g}) \sim N(\mu_x, \Sigma_x)$$

Where,

$$\mu_x = \Sigma_0 \Phi^T (\lambda I + \Phi \Sigma_0 \Phi^T)^{-1} y$$

$$\Sigma_x = \Sigma_0 - \Sigma_0 \Phi^T (\lambda I + \Phi \Sigma_0 \Phi^T)^{-1} \Phi \Sigma_0$$

μ_x, i.e. the mean of the posterior can be perceived as the point estimate of x.

BSBL framework

All parameters can be estimated by maximizing the Type II likelihood:

$$
\begin{aligned}
L(\Theta) &= -2 \log \int P(y|x;\lambda) P(x;(\gamma_i, B_i)_{i=1}^{g}) dx \\
&= \log |\lambda I + \Phi \Sigma_0 \Phi^T| + y^T (\lambda I + \Phi \Sigma_0 \Phi^T)^{-1} y
\end{aligned}
$$

Different optimization strategies lead to different BSBL algorithms.

BSBL Framework

BSBL-EM

Minimize the cost function using Expectation-Maximization.

BSBL-BO

Minimize the cost function using Bound Optimization technique (Majorize-Minimization).

BSBL-ℓ_1

Minimize the cost function using a sequence of reweighted ℓ_1 problems.

Summary

- Bayesian Methods offer Interesting Algorithmic Options
  - MAP estimation
  - 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.