Context & Notations

We consider the following $\ell_0$ penalized least squares problem,

$$
x = \text{arg min}_{x \in \mathbb{R}^M} G_0(x) := \frac{1}{2} \|Ax - d\|^2_2 + \lambda \|x\|_0,
$$

where $A \in \mathbb{R}^{M \times N}$, $M < N$, $d \in \mathbb{R}^M$. This NP-hard combinatorial problem is encountered in signal processing areas as coding, compressed sensing, variable selection, source separation, learning...

In this work, we propose a new continuous nonsmooth nonconvex penalty allowing an exact reformulation of problem (1).

Notations:
- $I_N = \{1, \ldots, N\}$, $s_i \in \mathbb{R}^N$ denotes the $i$th column of $A \in \mathbb{R}^{M \times N}$,
- $\sigma(x) := \{i \in I_N; x_i \neq 0\}$ defines the support of $x \in \mathbb{R}^N$,
- $\sigma^-(x) := \{i \in \sigma(x); |x_i| < \sqrt{\lambda}/|a_i|\}$ a part of the support.

The Continuous Exact $\ell_0$ (CEL0) penalty [6]

Replacing the $\ell_0$ pseudo norm in (1) by the CEL0 penalty,

$$
\Phi_{CEL0}(x) = \lambda \sqrt{\lambda} \left( \sum_{i \in \sigma^-(x)} |x_i|^2 \right)^{\frac{1}{2}},
$$

leads to a tight continuous relaxation of the objective function $G_0$ denoted $G_{CEL0}$.

(i) $G_{CEL0}$ is a global minimizer of $G_0$.

(ii) Conversely if $x \in \mathbb{R}^N$ is a global minimizer of $G_{CEL0}$, $x \in \mathbb{R}^N$ defined by

$$
\forall i \in I_N, \quad x_i^* = \begin{cases} 1 & \text{if } |x_i| > \sqrt{\lambda}/|a_i| \\ 0 & \text{otherwise} \end{cases}
$$

is also a global minimizer of $G_0$ via simple thresholding.

Links between minimizers of $G_{CEL0}$ and $G_{\ell_0}$ (for any $A \in \mathbb{R}^{M \times N}$)

Theorem 1 (link between global minimizers)

(i) the set of global minimizers of $G_{CEL0}$ is included in the set of global minimizers of $G_{\ell_0}$:

$$\text{arg min}_{x \in \mathbb{R}^N} G_{CEL0}(x) \subseteq \text{arg min}_{x \in \mathbb{R}^N} G_{\ell_0}(x)$$

(ii) conversely if $x \in \mathbb{R}^N$ is a global minimizer of $G_{CEL0}$, $x \in \mathbb{R}^N$ defined by

$$
\forall i \in I_N, \quad x_i^* = \begin{cases} 1 & \text{if } |x_i| > \sqrt{\lambda}/|a_i| \\ 0 & \text{otherwise} \end{cases}
$$

is a global minimizer of $G_{CEL0}$ and $G_{CEL0}(x^*) = G_{\ell_0}(x^*)$.

Numerical Illustrations

Fig. 2: Behaviour of $G_{CEL0}$ and $G_{\ell_0}$ for two examples where $N = M = 2$. Green points are local minimizers and red ones are global minimizers.

From [4], we know that a strict (local) minimizer of $G_{\ell_0}$ is easy to compute by choosing a support $\omega \in G_{CEL0}$ and solving the normal equations restricted to this support where,

$$
\omega_{CEL0} = \arg \min_{\omega \subseteq I_N} \|Ax - d\|_2^2
$$

Hence, for a given matrix $A$ and vector $d$, one can compute all the strict (local) minimizers of $G_{CEL0}$ since $G_{CEL0}$ is a finite set and check which ones are also critical points for $G_{CEL0}$ (i.e. verify the conditions of (6). Lemma 4.1).

Minimizing $G_{CEL0}$

The continuity of $G_{CEL0}$ allows to use nonsmooth nonconvex algorithms for its minimization,

- DC programming [3]
- MM algorithms (e.g. IRL1 [5])
- Forward-Backward splitting [1]

Lemma (critical points of $G_{CEL0}$ and minimizers of $G_{\ell_0}$)

Let $s \in \mathbb{R}^N$ be a critical point of $G_{CEL0}$ verifying $\sigma^-(s) = \emptyset$. Then it is a (local) minimizer of $G_{CEL0}$ and $G_{CEL0}(s) = G_{\ell_0}(s)$.

Macro algorithm [6, Algo 1]
- adds an outer loop to any nonsmooth nonconvex algorithm, e.g. see (Algo).
- moves iteratively from a critical point $s$ of $G_{CEL0}$ to another one, while decreasing the cost function, until $\sigma^-(s) = \emptyset$
- converges to a minimizer of $G_{CEL0}$ [6, Th 5.1]

Links between minimizers of $G_{CEL0}$ and $G_{\ell_0}$ (for any $A \in \mathbb{R}^{M \times N}$)

Fig. 4: 2D-illustrations of the Macro algorithm using IRL1 (left) and FB (right)

Fig. 5: Performance comparison between the Macro algorithm and the Forward-Backward Splitting (FB) algorithm [2, 1]. Cumulative histograms of the error $|\hat{x} - x^*|_2$ where $\hat{x}$ is the estimated solution and $x^*$ is a global minimizer of $G_0$. The histograms are computed from 1000 random matrices $A$ and $d \in \mathbb{R}^N$ generated from a uniform distribution. From left to right: $\lambda = 0.1$, $\lambda = 0.5$ and $\lambda = 1$. Top line: minimization $s_0 = A^T d$. Bottom line: minimization $s_0 = s_0^*$.  

References


