On the accuracy of ℓ_1-filtering of signals with block-sparse structure

A. Juditsky[1], F. Kilinc Karzan[2], A. Nemirovski[3] and Boris Polyak[4]

[1]Université J. Fourier de Grenoble
[2]Carnegie Mellon University
[3]Georgia Institute of Technology
[4]Institute of Control Sciences

Block-sparse recovery problem

Given $y = Ax + u + \xi$, $y \in \mathbf{R}^m$ recover $w = Bx \in \mathbf{R}^N$, where

- $A \in \mathbf{R}^{m \times n}$ is the *sensing matrix*;
- $\xi \in \mathbf{R}^m$ is the observation noise, $\xi \sim \mathcal{N}(0, D)$, $D \succeq 0$ is known;
- $u \in \mathcal{U}$ is an unknown *nuisance signal*, though $\mathcal{U}$ is a known set
- $B \in \mathbf{R}^{n \times N}$

A priori information available:

> *we assume that w is a block-vector: $w = [w[1]; \ldots.; w[K]]$ with blocks $w[k] \in \mathbf{R}^{n_k}$ and that w is almost block-sparse:*
> it is "well approximated" most with a block-vector w^s such that only a given number $s < K$ of blocks $w[k], 1 \leq k \leq K$, does not vanish.

Block-ℓ_1 recovery

Given an $\epsilon > 0$ and an $m \times N$ contrast matrix $H = [h^1, ..., h^N]$, we introduce two recovery routines:

- *regular L_1 recovery* (cf. (block-) Dantzig selector)

$$\widehat{x}_{\mathrm{reg}}(y) \in \operatorname*{Argmin}_{z \in \mathbb{R}^n} \left\{ L_1(Bz) : \ \|H^T(y - Az)\|_\infty \leq \nu(H) \right\},$$

  and

- *penalized L_1 recovery (cf. (block-) Lasso)*

$$\widehat{x}_{\mathrm{pen}}(y) \in \operatorname*{Argmin}_{z \in \mathbb{R}^n} \left[L_1(Bz) + \kappa \|H^T(y - Az)\|_\infty \right].$$

Questions:

- given a sensing matrix A, how can one verify that block-ℓ_1 recovery "makes sense", e.g., reproduces block-sparse signals w with a "small" error?
- can one provide confidence sets for these recoveries, i.e. compute certifiable accuracy bounds for the proposed procedures?
- is it possible to choose the contrast matrix H to attain the best possible accuracy bounds?
- what can be said about the optimality of the proposed procedures?
- what is the "numerical performance" of these algorithms?