Node-screening tests for the ℓ_0-penalized least-squares problem

Theo Guyard*, Cedric Herzet†, Clement Elvira‡

ICASSP 2022

* Univ Rennes, INSA Rennes, CNRS, IRMAR-UMR 6625, F-35000 Rennes, France
† INRIA Rennes-Bretagne Atlantique, Campus de Beaulieu, 35000 Rennes, France
‡ SCEE/IETR UMR CNRS 6164, CentraleSupélec, 35510 Cesson Sévigné, France

Table of contents

The ℓ_0-penalized least-squares problem

Sparse-linear problem

Ingredients of the problem :

- An observation $y \in \mathbb{R}^m$
- A dictionary $A = [a_i]_{i=1}^n \in \mathbb{R}^{m \times n}$ (columns $\equiv$ atoms)

Sparse-linear problem

Ingredients of the problem :

- An observation $y \in \mathbb{R}^m$
- A dictionary $A = [a_i]_{i=1}^n \in \mathbb{R}^{m \times n}$ (columns $\equiv$ atoms)

Objectives :

- Approximate the observation as a linear combination of the atoms
- The linear combination must be *sparse*

Sparse-linear problem

Ingredients of the problem :

- An observation $y \in \mathbb{R}^m$
- A dictionary $A = [a_i]_{i=1}^n \in \mathbb{R}^{m \times n}$ (columns $\equiv$ atoms)

Objectives :

- Approximate the observation as a linear combination of the atoms
- The linear combination must be *sparse*

Problem

Find x *sparse* such that $y \simeq Ax$

The vector x weights each atom in the approximation.

ℓ_0-penalized problem

Idea : Solve the problem

ℓ_0-penalized least-squares

$$p^\star = \begin{cases} \min & \frac{1}{2}\|y - Ax\|_2^2 + \lambda\|x\|_0 \\ \text{s.t.} & \|x\|_\infty \leq M \end{cases} \qquad (P)$$

where $\lambda > 0$ is a tuning parameter and M is a big-enough constant.

ℓ_0-penalized problem

Idea : Solve the problem

ℓ_0-penalized least-squares

$$p^\star = \begin{cases} \min & \frac{1}{2}\|y - Ax\|_2^2 + \lambda\|x\|_0 \\ \text{s.t.} & \|x\|_\infty \leq M \end{cases} \quad (P)$$

where $\lambda > 0$ is a tuning parameter and M is a big-enough constant.

$$\text{Problem } (P) \xrightarrow{\text{reformulation}} \text{Mixed-Integer Program}$$

ℓ_0-penalized problem

Idea : Solve the problem

ℓ_0-penalized least-squares

$$p^\star = \begin{cases} \min & \frac{1}{2}\|y - Ax\|_2^2 + \lambda\|x\|_0 \\ \text{s.t.} & \|x\|_\infty \leq M \end{cases} \qquad (P)$$

where $\lambda > 0$ is a tuning parameter and M is a big-enough constant.

$$\text{Problem } (P) \xrightarrow{\text{reformulation}} \text{Mixed-Integer Program}$$

Properties :

- Quadratic objective
- Linear constraints
- Continuous and integer variables
- Combinatorial problem
- Can be addressed with Branch-and-Bound (BnB) algorithms

Branch-and-bound algorithms

Branch-and-bound principle

Idea :

- Enumerate all feasible solutions
- Use rules to discard irrelevant candidates

$\rightarrow$ In a nutshell : explore a decision tree and prune uninteresting nodes

Branch-and-bound principle

Idea :

- Enumerate all feasible solutions
- Use rules to discard irrelevant candidates

→ In a nutshell : explore a decision tree and prune uninteresting nodes

Node $\nu = (\mathcal{S}_0, \mathcal{S}_1, \bar{\mathcal{S}})$ **where :**

- $\mathcal{S}_0$: indices of x fixed to zero
- $\mathcal{S}_1$: indices of x fixed to non-zero
- $\bar{\mathcal{S}}$: indices not fixed yet

Branch-and-bound principle

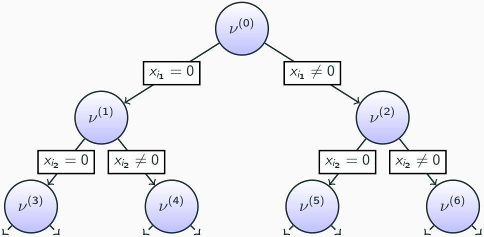
Idea :

- Enumerate all feasible solutions
- Use rules to discard irrelevant candidates

$\rightarrow$ In a nutshell : explore a decision tree and prune uninteresting nodes

Node $\nu = (\mathcal{S}_0, \mathcal{S}_1, \bar{\mathcal{S}})$ **where :**

- $\mathcal{S}_0$: indices of x fixed to zero
- $\mathcal{S}_1$: indices of x fixed to non-zero
- $\bar{\mathcal{S}}$: indices not fixed yet



Processing node $\nu = (\mathcal{S}_0, \mathcal{S}_1, \bar{\mathcal{S}})$

Question : Does any global solution matches the current constraints ?

Processing node $\nu = (\mathcal{S}_0, \mathcal{S}_1, \bar{\mathcal{S}})$

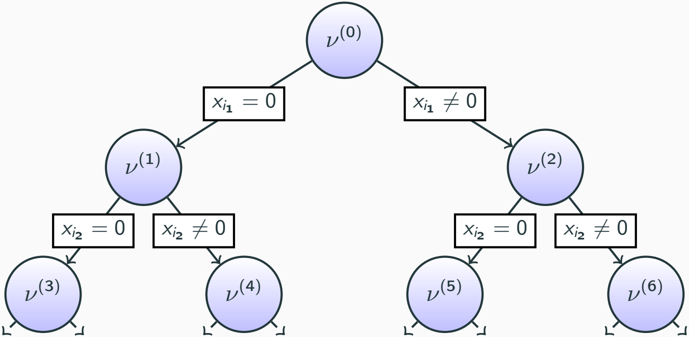
Question : Does any global solution matches the current constraints ?

Relaxed problem at node ν

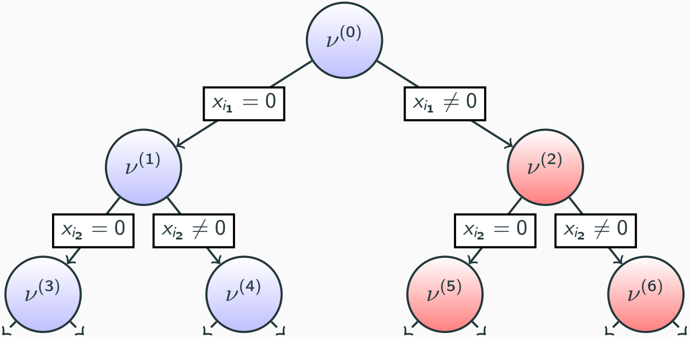
$$p_l^\nu \;=\begin{cases} \min & \frac{1}{2}\|\mathrm{y} - \mathrm{Ax}\|_2^2 + \frac{\lambda}{M}\|\mathrm{x}_{\bar{\mathcal{S}}}\|_1 + \lambda|\mathcal{S}_1| \\ \text{s.t.} & \|\mathrm{x}\|_\infty \leq M,\; \mathrm{x}_{\mathcal{S}_0} = 0 \end{cases} \qquad (P_l^\nu)$$

Let p_u be an upper bound on $p^\star$. If $p_u < p_l^\nu$, then no optimizers of (P) can match the constraints of node ν.

Exploration and pruning process



Exploration and pruning process



Node-screening tests

Dual problem

Question : Is it possible to detect nodes that cannot yield a global optimizers without processing them ?

Dual problem

Question : Is it possible to detect nodes that cannot yield a global optimizers without processing them ?

Dual problem at node ν

$$\max_{\mathbf{u}\in\mathbb{R}^m} \left\{ \mathrm{D}^{\nu}(\mathbf{u}) \triangleq \frac{1}{2}\|\mathbf{y}\|_2^2 - \frac{1}{2}\|\mathbf{y}-\mathbf{u}\|_2^2 - \sum_{i\in\bar{\mathcal{S}}}[\gamma(\mathbf{a}_i^{\mathsf{T}}\mathbf{u})]_+ - \sum_{i\in\mathcal{S}_1}\gamma(\mathbf{a}_i^{\mathsf{T}}\mathbf{u}) \right\} \quad (D^{\nu})$$

Dual problem

Question : Is it possible to detect nodes that cannot yield a global optimizers without processing them ?

Dual problem at node ν

$$\max_{\mathrm{u}\in\mathbb{R}^m} \left\{ \mathrm{D}^{\nu}(\mathrm{u}) \triangleq \tfrac{1}{2}\|\mathrm{y}\|_2^2 - \tfrac{1}{2}\|\mathrm{y}-\mathrm{u}\|_2^2 - \sum_{i\in\bar{\mathcal{S}}}[\gamma(\mathrm{a}_i^{\mathrm{T}}\mathrm{u})]_+ - \sum_{i\in\mathcal{S}_1}\gamma(\mathrm{a}_i^{\mathrm{T}}\mathrm{u}) \right\} \quad (D^{\nu})$$

- One common term
- Terms corresponding to the current constraints
- The "pivot" function is defined as $\gamma(t) = M|t| - \lambda$

Dual objective link

Direct consequence : The objective of two consecutive nodes differs from one term.

Dual objective link

Direct consequence : The objective of two consecutive nodes differs from one term.

Dual objective link

At node ν, let i be an unfixed index. Then $\forall \mathsf{u} \in \mathbb{R}^m$,

$$\mathrm{D}^{\nu \cup \{x_i = 0\}}(\mathsf{u}) = \mathrm{D}^{\nu}(\mathsf{u}) + [\gamma(\mathsf{a}_i^\mathsf{T} \mathsf{u})]_+$$
$$\mathrm{D}^{\nu \cup \{x_i \neq 0\}}(\mathsf{u}) = \mathrm{D}^{\nu}(\mathsf{u}) + [\gamma(\mathsf{a}_i^\mathsf{T} \mathsf{u})]_-$$

Dual objective link

Direct consequence : The objective of two consecutive nodes differs from one term.

Dual objective link

At node ν, let i be an unfixed index. Then $\forall \mathsf{u} \in \mathbb{R}^m$,

$$\mathrm{D}^{\nu \cup \{x_i = 0\}}(\mathsf{u}) = \mathrm{D}^{\nu}(\mathsf{u}) + [\gamma(\mathsf{a}_i^\mathsf{T} \mathsf{u})]_+$$
$$\mathrm{D}^{\nu \cup \{x_i \neq 0\}}(\mathsf{u}) = \mathrm{D}^{\nu}(\mathsf{u}) + [\gamma(\mathsf{a}_i^\mathsf{T} \mathsf{u})]_-$$

- $\forall \mathsf{u}$, $\mathrm{D}^{\nu}(\mathsf{u}) \leq p_l^{\nu}$: the dual objective can also be used to prune nodes.
- At a given node, we may be able to prune subnodes without processing them.

Node-screening test

Node-screening test

Given an upper bound p_u on $p^\star$ and a dual point $\mathrm{u} \in \mathbb{R}^m$,

$$\mathrm{D}^\nu(\mathrm{u}) + [\gamma(\mathrm{a}_i^\mathsf{T}\mathrm{u})]_+ > p_u \quad \Longrightarrow \quad \text{Fix } x_i \neq 0 \text{ at node } \nu$$
$$\mathrm{D}^\nu(\mathrm{u}) + [\gamma(\mathrm{a}_i^\mathsf{T}\mathrm{u})]_- > p_u \quad \Longrightarrow \quad \text{Fix } x_i = 0 \text{ at node } \nu$$

Node-screening test

Node-screening test

Given an upper bound p_u on $p^\star$ and a dual point $\mathrm{u} \in \mathbb{R}^m$,

$$\begin{aligned} \mathrm{D}^\nu(\mathrm{u}) + [\gamma(\mathrm{a}_i^\mathsf{T}\mathrm{u})]_+ > p_u \quad &\Longrightarrow \quad \text{Fix } x_i \neq 0 \text{ at node } \nu \\ \mathrm{D}^\nu(\mathrm{u}) + [\gamma(\mathrm{a}_i^\mathsf{T}\mathrm{u})]_- > p_u \quad &\Longrightarrow \quad \text{Fix } x_i = 0 \text{ at node } \nu \end{aligned}$$

Practical use : If a node-screening test is passed at node ν, one can immediately fix a new variable at this node.

Node-screening test

Node-screening test

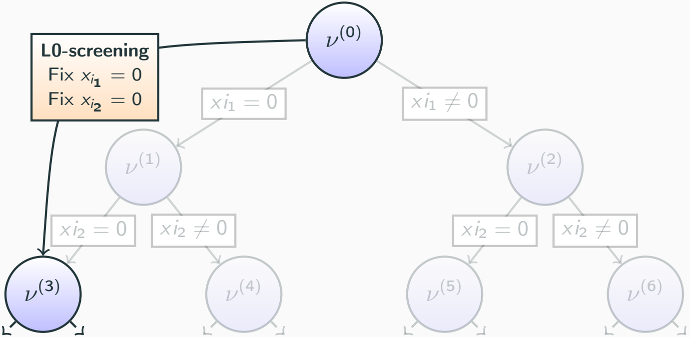
Given an upper bound p_u on $p^\star$ and a dual point $\mathrm{u} \in \mathbb{R}^m$,

$$\begin{aligned} \mathrm{D}^\nu(\mathrm{u}) + [\gamma(\mathrm{a}_i^\mathsf{T}\mathrm{u})]_+ > p_u \quad &\Longrightarrow \quad \text{Fix } x_i \neq 0 \text{ at node } \nu \\ \mathrm{D}^\nu(\mathrm{u}) + [\gamma(\mathrm{a}_i^\mathsf{T}\mathrm{u})]_- > p_u \quad &\Longrightarrow \quad \text{Fix } x_i = 0 \text{ at node } \nu \end{aligned}$$

Practical use : If a node-screening test is passed at node ν, one can immediately fix a new variable at this node.

Nesting property : If multiple node-screening tests are passed, the corresponding variables can be fixed simultaneously.

Consequence of passing a node-screening test



Consequence : Less nodes are explored by the BnB algorithm.

Some numerical results

Some numerical results

Synthetic setups :

1. Generate the dictionary randomly (low or high correlation)
2. Generate a k-sparse vector $\mathrm{x}^\star$
3. Set $\mathrm{y} = \mathrm{A}\mathrm{x}^\star +$ noise
4. Tune λ and M to (hopefully) recover $\mathrm{x}^\star$ by solving (P)

Some numerical results

Synthetic setups :

1. Generate the dictionary randomly (low or high correlation)
2. Generate a k-sparse vector $x^\star$
3. Set $y = Ax^\star +$ noise
4. Tune λ and M to (hopefully) recover $x^\star$ by solving (P)

Methods compared : CPLEX (commercial solver), a tailored BnB and a tailored BnB with node-screening tests.

Some numerical results

Synthetic setups :

1. Generate the dictionary randomly (low or high correlation)
2. Generate a k-sparse vector $\mathrm{x}^\star$
3. Set $\mathrm{y} = \mathrm{A}\mathrm{x}^\star +$ noise
4. Tune λ and M to (hopefully) recover $\mathrm{x}^\star$ by solving (P)

Methods compared : CPLEX (commercial solver), a tailored BnB and a tailored BnB with node-screening tests.

		CPLEX		BnB		BnB+scr	
Corr.	**Sparsity**	Nodes	Time	Nodes	Time	**Nodes**	**Time**
Low	$k=3$	16	13.13	19	0.29	**15**	**0.18**
	$k=5$	96	25.89	70	1.5	**56**	**0.75**
	$k=7$	292	60.84	180	5.14	**152**	**3.02**
High	$k=3$	76	1.73	79	0.38	**60**	**0.26**
	$k=5$	1,424	10.18	965	6.39	**725**	**4.18**
	$k=7$	17,647	106.45	10,461	79.29	**7,881**	**52.16**