Node-screening pour le problème des moindres carrés avec pénalité ℓ_0

Théo Guyard[1,2], Ayse-Nur Arslan[1], Cédric Herzet[2], Clément Elvira[3]
[1] INSA Rennes
[2] INRIA Rennes Bretagne Atlantique
[3] IETR CentraleSupélec

GRETSI – 8 Septembre 2022

ℓ_0-penalized problems

Sparse problem

Ingredients of the problem

- A target y

Sparse problem

Ingredients of the problem

- A target y
- A dictionary $A = \{a_i\}_{i \in \mathcal{I}}$ made of atoms

Sparse problem

Ingredients of the problem

- A target y
- A dictionary $\mathsf{A} = \{\mathsf{a}_i\}_{i \in \mathcal{I}}$ made of atoms

Objective

- Find a sparse linear combination of atoms that well approximates the target through a given model

Sparse problem

Ingredients of the problem

- A target y
- A dictionary $A = \{a_i\}_{i \in \mathcal{I}}$ made of atoms

Objective

- Find a sparse linear combination of atoms that well approximates the target through a given model

Rough formulation

Problem

Find x sparse such that $y \simeq \text{Model}(Ax)$

Sparse problem

Ingredients of the problem

- A target y
- A dictionary $\mathrm{A} = \{\mathrm{a}_i\}_{i \in \mathcal{I}}$ made of atoms

Objective

- Find a sparse linear combination of atoms that well approximates the target through a given model

Rough formulation

Problem

Find x sparse such that $\mathrm{y} \simeq \mathrm{Model}(\mathrm{Ax})$

Remark : Entries of x weight each atom in the linear combination.

ℓ_0-penalized problem

ℓ_0-penalized problem

$$p^\star = \begin{cases} \min & f(\mathrm{A}\mathrm{x}) + \lambda \|\mathrm{x}\|_0 \\ \text{s.t.} & \|\mathrm{x}\|_\infty \leq M \end{cases} \tag{P}$$

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

ℓ_0-penalized problem

ℓ_0-penalized problem

$$p^\star = \begin{cases} \min & f(\mathrm{A}\mathrm{x}) + \lambda\|\mathrm{x}\|_0 \\ \text{s.t.} & \|\mathrm{x}\|_\infty \leq M \end{cases} \tag{P}$$

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

ℓ_0-penalized problem

$$p^\star = \begin{cases} \min & f(\mathbf{A}\mathbf{x}) + \lambda\|\mathbf{x}\|_0 \\ \text{s.t.} & \|\mathbf{x}\|_\infty \leq M \end{cases} \tag{P}$$

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 :

- 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

Branch-and-bound principle

Idea :

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

Branch-and-bound principle

Idea :

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

$\rightarrow$ Explore a decision tree and prune uninteresting nodes

Branch-and-bound principle

Idea :

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

→ 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

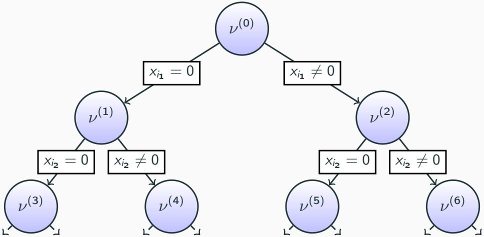
Idea :

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

$\rightarrow$ 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 solution of (P) match the current constraints ?

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

Question : Does any solution of (P) match the current constraints ?

Sub-problem at node ν

$$p^\nu = \left\{ \begin{array}{ll} \min & f(\mathbf{A}\mathbf{x}) + \lambda\|\mathbf{x}\|_0 \\ \text{s.t.} & \|\mathbf{x}\|_\infty \leq M \end{array} \right\} \bigcap \left\{ \begin{array}{ll} \mathbf{x}_{\mathcal{S}_0} & = 0 \\ \mathbf{x}_{\mathcal{S}_1} & \neq 0 \end{array} \right\} \qquad (P^\nu)$$

If $p^\star < p^\nu$, then node ν can be pruned from the BnB tree.

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

Question : Does any solution of (P) match the current constraints ?

Sub-problem at node ν

$$p^\nu = \left\{ \begin{array}{ll} \min & f(\mathrm{A}\mathrm{x}) + \lambda\|\mathrm{x}\|_0 \\ \text{s.t.} & \|\mathrm{x}\|_\infty \leq M \end{array} \right\} \bigcap \left\{ \begin{array}{ll} \mathrm{x}_{\mathcal{S}_0} & = 0 \\ \mathrm{x}_{\mathcal{S}_1} & \neq 0 \end{array} \right\} \qquad (P^\nu)$$

If $p^\star < p^\nu$, then node ν can be pruned from the BnB tree.

Neither $p^\star$ nor p^ν are accessible in practice :

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

Question : Does any solution of (P) match the current constraints ?

Sub-problem at node ν

$$p^\nu = \left\{ \begin{array}{ll} \min & f(\mathrm{Ax}) + \lambda\|\mathrm{x}\|_0 \\ \text{s.t.} & \|\mathrm{x}\|_\infty \leq M \end{array} \right\} \bigcap \left\{ \begin{array}{ll} \mathrm{x}_{\mathcal{S}_0} & = 0 \\ \mathrm{x}_{\mathcal{S}_1} & \neq 0 \end{array} \right\} \tag{P^ν}$$

If $p^\star < p^\nu$, then node ν can be pruned from the BnB tree.

Neither $p^\star$ nor p^ν are accessible in practice :

- Restriction : Upper bound $\bar{p}$ on $p^\star$

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

Question : Does any solution of (P) match the current constraints ?

Sub-problem at node ν

$$p^\nu = \left\{ \begin{array}{ll} \min & f(\mathrm{Ax}) + \lambda \|\mathrm{x}\|_0 \\ \text{s.t.} & \|\mathrm{x}\|_\infty \leq M \end{array} \right\} \bigcap \left\{ \begin{array}{ll} \mathrm{x}_{\mathcal{S}_0} & = 0 \\ \mathrm{x}_{\mathcal{S}_1} & \neq 0 \end{array} \right\} \qquad (P^\nu)$$

If $p^\star < p^\nu$, then node ν can be pruned from the BnB tree.

Neither $p^\star$ nor p^ν are accessible in practice :

- Restriction : Upper bound $\bar{p}$ on $p^\star$
- Relaxation : Lower bound $\tilde{p}^\nu$ on p^ν

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

Question : Does any solution of (P) match the current constraints ?

Sub-problem at node ν

$$p^\nu = \left\{ \begin{array}{ll} \min & f(\mathrm{Ax}) + \lambda\|\mathrm{x}\|_0 \\ \text{s.t.} & \|\mathrm{x}\|_\infty \leq M \end{array} \right\} \bigcap \left\{ \begin{array}{ll} \mathrm{x}_{\mathcal{S}_0} & = 0 \\ \mathrm{x}_{\mathcal{S}_1} & \neq 0 \end{array} \right\} \qquad (P^\nu)$$

If $p^\star < p^\nu$, then node ν can be pruned from the BnB tree.

Neither $p^\star$ nor p^ν are accessible in practice :

- Restriction : Upper bound $\bar{p}$ on $p^\star$
- Relaxation : Lower bound $\tilde{p}^\nu$ on p^ν
- If $\bar{p} < \tilde{p}^\nu$, then node ν can be pruned

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

Question : Does any solution of (P) match the current constraints ?

Sub-problem at node ν

$$p^\nu = \left\{ \begin{array}{cc} \min & f(\mathrm{Ax}) + \lambda\|\mathrm{x}\|_0 \\ \text{s.t.} & \|\mathrm{x}\|_\infty \leq M \end{array} \right\} \bigcap \left\{ \begin{array}{cc} \mathrm{x}_{\mathcal{S}_0} & = 0 \\ \mathrm{x}_{\mathcal{S}_1} & \neq 0 \end{array} \right\} \qquad (P^\nu)$$

If $p^\star < p^\nu$, then node ν can be pruned from the BnB tree.

Neither $p^\star$ nor p^ν are accessible in practice :

- Restriction : Upper bound $\bar{p}$ on $p^\star$
- Relaxation : Lower bound $\tilde{p}^\nu$ on p^ν
- If $\bar{p} < \tilde{p}^\nu$, then node ν can be pruned
- Both $\bar{p}$ and $\tilde{p}^\nu$ are computed by solving convex problems

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

Question : Does any solution of (P) match the current constraints ?

Sub-problem at node ν

$$p^\nu = \left\{ \begin{array}{ll} \min & f(\mathrm{Ax}) + \lambda\|\mathrm{x}\|_0 \\ \text{s.t.} & \|\mathrm{x}\|_\infty \leq M \end{array} \right\} \bigcap \left\{ \begin{array}{ll} \mathrm{x}_{\mathcal{S}_0} & = 0 \\ \mathrm{x}_{\mathcal{S}_1} & \neq 0 \end{array} \right\} \tag{P^ν}$$

If $p^\star < p^\nu$, then node ν can be pruned from the BnB tree.

Neither $p^\star$ nor p^ν are accessible in practice :

- Restriction : Upper bound $\bar{p}$ on $p^\star$
- Relaxation : Lower bound $\tilde{p}^\nu$ on p^ν
- If $\bar{p} < \tilde{p}^\nu$, then node ν can be pruned
- Both $\bar{p}$ and $\tilde{p}^\nu$ are computed by solving convex problems

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

Question : Does any solution of (P) match the current constraints ?

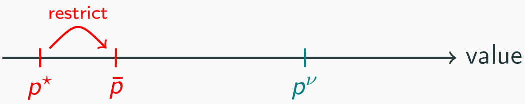
Sub-problem at node ν

$$p^\nu = \left\{ \begin{array}{ll} \min & f(\mathrm{A}\mathrm{x}) + \lambda\|\mathrm{x}\|_0 \\ \text{s.t.} & \|\mathrm{x}\|_\infty \leq M \end{array} \right\} \bigcap \left\{ \begin{array}{ll} \mathrm{x}_{\mathcal{S}_0} & = 0 \\ \mathrm{x}_{\mathcal{S}_1} & \neq 0 \end{array} \right\} \qquad (P^\nu)$$

If $p^\star < p^\nu$, then node ν can be pruned from the BnB tree.

Neither $p^\star$ nor p^ν are accessible in practice :

- Restriction : Upper bound $\bar{p}$ on $p^\star$
- Relaxation : Lower bound $\tilde{p}^\nu$ on p^ν
- If $\bar{p} < \tilde{p}^\nu$, then node ν can be pruned
- Both $\bar{p}$ and $\tilde{p}^\nu$ are computed by solving convex problems



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

Question : Does any solution of (P) match the current constraints ?

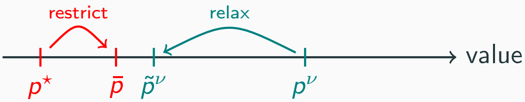
Sub-problem at node ν

$$p^{\nu} = \left\{ \begin{array}{ll} \min & f(\mathrm{A}\mathrm{x}) + \lambda\|\mathrm{x}\|_0 \\ \text{s.t.} & \|\mathrm{x}\|_\infty \leq M \end{array} \right\} \bigcap \left\{ \begin{array}{ll} \mathrm{x}_{\mathcal{S}_0} & = 0 \\ \mathrm{x}_{\mathcal{S}_1} & \neq 0 \end{array} \right\} \qquad (P^{\nu})$$

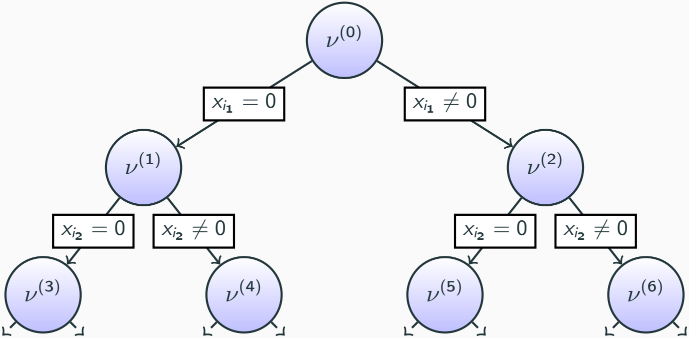
If $p^{\star} < p^{\nu}$, then node ν can be pruned from the BnB tree.

Neither $p^{\star}$ nor p^{ν} are accessible in practice :

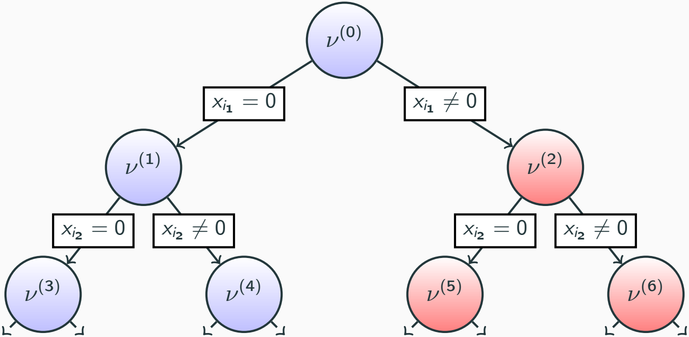
- Restriction : Upper bound $\bar{p}$ on $p^{\star}$
- Relaxation : Lower bound $\tilde{p}^{\nu}$ on p^{ν}
- If $\bar{p} < \tilde{p}^{\nu}$, then node ν can be pruned
- Both $\bar{p}$ and $\tilde{p}^{\nu}$ are computed by solving convex problems



Exploration and pruning process



Exploration and pruning process



BnB efficiency

The efficiency of the BnB algorithm depends on :

- The number of nodes processed
- The ability to process nodes quickly

Node-screening improves both of these things !

Node-screening

Main idea

Testing if a node can be pruned $\equiv$ Solving convex problems

Sometimes it is obvious !

This is costly

Main idea

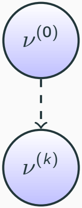
Testing if a node can be pruned $\equiv$ Solving convex problems

$\uparrow$ Sometimes it is obvious !

$\uparrow$ This is costly

Question : How to detect prunable nodes in a more economic way ?

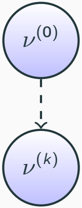
Dual problem



Node $\nu^{(k)} = (\mathcal{S}_{\mathbf{0}}, \mathcal{S}_{\mathbf{1}}, \bar{\mathcal{S}})$

- Sub-problem
- Relaxed problem

Dual problem



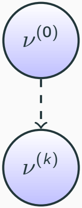
Node $\nu^{(k)} = (\mathcal{S}_{\mathbf{0}}, \mathcal{S}_{\mathbf{1}}, \bar{\mathcal{S}})$

- Sub-problem
- Relaxed problem

Dual problem at node ν

$$\max_{\mathbf{u}\in\mathbb{R}^m} \left\{ \mathrm{D}^{\nu}(\mathbf{u}) \triangleq -f^*(-\mathbf{u}) - \sum_{i\in\bar{\mathcal{S}}} [\gamma(\mathbf{a}_i^{\mathsf{T}}\mathbf{u})]_+ - \sum_{i\in\mathcal{S}_{\mathbf{1}}} \gamma(\mathbf{a}_i^{\mathsf{T}}\mathbf{u}) \right\} \qquad (D^{\nu})$$

Dual problem



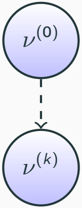
Node $\nu^{(k)} = (\mathcal{S}_\mathbf{0}, \mathcal{S}_\mathbf{1}, \bar{\mathcal{S}})$
- Sub-problem
- Relaxed problem

Dual problem at node ν

$$\max_{\mathrm{u} \in \mathbb{R}^m} \left\{ \mathrm{D}^\nu(\mathrm{u}) \triangleq -f^*(-\mathrm{u}) - \sum_{i \in \bar{\mathcal{S}}} [\gamma(\mathrm{a}_i^\mathsf{T} \mathrm{u})]_+ - \sum_{i \in \mathcal{S}_\mathbf{1}} \gamma(\mathrm{a}_i^\mathsf{T} \mathrm{u}) \right\} \qquad (D^\nu)$$

- One common term to all nodes
- Terms depending on the current node constraints

Dual problem



Node $\nu^{(k)} = (\mathcal{S}_{\mathbf{0}}, \mathcal{S}_{\mathbf{1}}, \bar{\mathcal{S}})$
- Sub-problem
- Relaxed problem

Dual problem at node ν

$$\max_{\mathrm{u}\in\mathbb{R}^m} \left\{ \mathrm{D}^{\nu}(\mathrm{u}) \triangleq -f^*(-\mathrm{u}) - \sum_{i\in\bar{\mathcal{S}}} [\gamma(\mathrm{a}_i^{\mathsf{T}}\mathrm{u})]_+ - \sum_{i\in\mathcal{S}_{\mathbf{1}}} \gamma(\mathrm{a}_i^{\mathsf{T}}\mathrm{u}) \right\} \qquad (D^{\nu})$$

- One common term to all nodes
- Terms depending on the current node constraints
- The pivot function is defined as $\gamma(t) = M|t| - \lambda$

Dual objective link

Direct consequence : The dual objective at two consecutive nodes differs from only one term.

Dual objective link

Direct consequence : The dual objective at two consecutive nodes differs from only one term.

Dual objective link

At node $\nu = (\mathcal{S}_0, \mathcal{S}_1, \bar{\mathcal{S}})$, let $i \in \bar{\mathcal{S}}$. Then $\forall \mathsf{u}$,

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

Dual objective link

Direct consequence : The dual objective at two consecutive nodes differs from only one term.

Dual objective link

At node $\nu = (\mathcal{S}_0, \mathcal{S}_1, \bar{\mathcal{S}})$, let $i \in \bar{\mathcal{S}}$. Then $\forall \mathsf{u}$,

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

Dual objective link

Direct consequence : The dual objective at two consecutive nodes differs from only one term.

Dual objective link

At node $\nu = (\mathcal{S}_0, \mathcal{S}_1, \bar{\mathcal{S}})$, let $i \in \bar{\mathcal{S}}$. Then $\forall \mathsf{u}$,

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

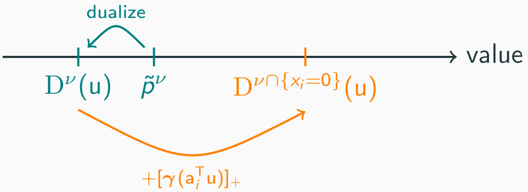
Dual objective link

Direct consequence : The dual objective at two consecutive nodes differs from only one term.

Dual objective link

At node $\nu = (\mathcal{S}_0, \mathcal{S}_1, \bar{\mathcal{S}})$, let $i \in \bar{\mathcal{S}}$. Then $\forall \mathbf{u}$,

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



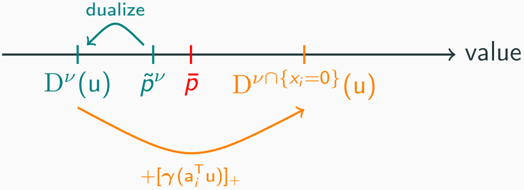
Dual objective link

Direct consequence : The dual objective at two consecutive nodes differs from only one term.

Dual objective link

At node $\nu = (\mathcal{S}_0, \mathcal{S}_1, \bar{\mathcal{S}})$, let $i \in \bar{\mathcal{S}}$. Then $\forall \mathrm{u}$,

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



Node-screening test

Node-screening test

Given some point u,

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

Node-screening test

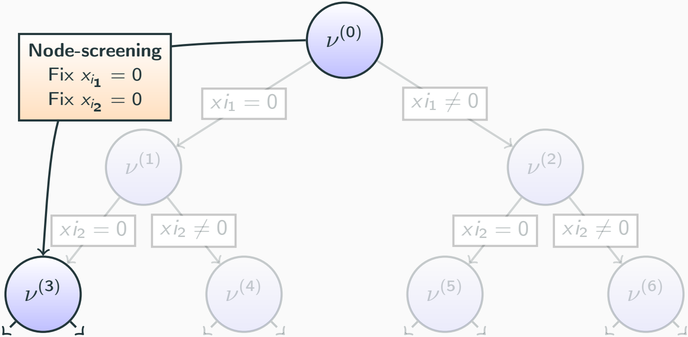
Node-screening test

Given some point u,

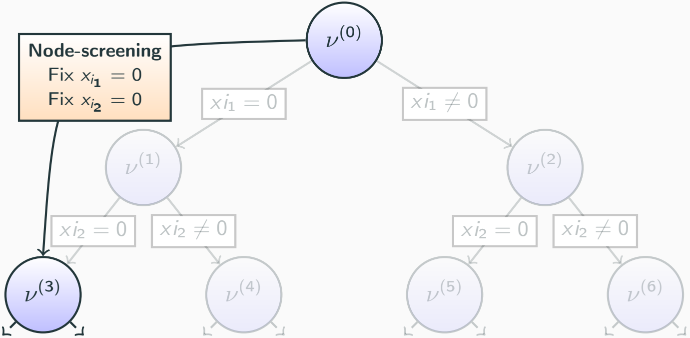
$$\mathrm{D}^{\nu}(\mathrm{u}) + [\gamma(\mathrm{a}_i^{\mathsf{T}}\mathrm{u})]_+ > \bar{p} \quad \Longrightarrow \quad \text{Fix } x_i \neq 0 \text{ at node } \nu$$
$$\mathrm{D}^{\nu}(\mathrm{u}) - [\gamma(\mathrm{a}_i^{\mathsf{T}}\mathrm{u})]_- > \bar{p} \quad \Longrightarrow \quad \text{Fix } x_i =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

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

Consequence of passing a node-screening test



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. Sample a synthetic $A \in \mathbb{R}^{m \times n}$ with $(m, n) = (500, 1000)$

Some numerical results

Synthetic setups :

1. Sample a synthetic $\mathrm{A} \in \mathbb{R}^{m \times n}$ with $(m, n) = (500, 1000)$
2. Generate a k-sparse vector $\mathrm{x}^\star$ with $k \in \{5, \dots, 20\}$

Some numerical results

Synthetic setups :

1. Sample a synthetic $\mathrm{A} \in \mathbb{R}^{m \times n}$ with $(m, n) = (500, 1000)$
2. Generate a k-sparse vector $\mathrm{x}^\star$ with $k \in \{5, \dots, 20\}$
3. Set $\mathrm{y} = \mathrm{A}\mathrm{x}^\star +$ noise with 10dB SNR

Some numerical results

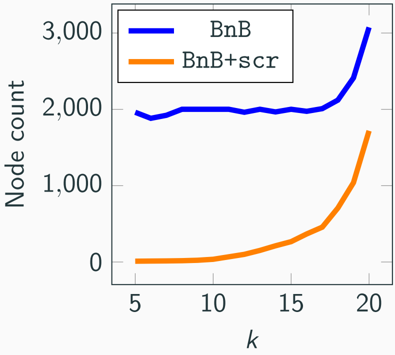
Synthetic setups :

1. Sample a synthetic $\mathrm{A} \in \mathbb{R}^{m \times n}$ with $(m, n) = (500, 1000)$
2. Generate a k-sparse vector $\mathrm{x}^\star$ with $k \in \{5, \dots, 20\}$
3. Set $\mathrm{y} = \mathrm{A}\mathrm{x}^\star +$ noise with 10dB SNR
4. Tune λ and M to (hopefully) recover $\mathrm{x}^\star$ by solving (P)

Some numerical results

Synthetic setups :

1. Sample a synthetic $\mathrm{A} \in \mathbb{R}^{m \times n}$ with $(m, n) = (500, 1000)$
2. Generate a k-sparse vector $\mathrm{x}^\star$ with $k \in \{5, \dots, 20\}$
3. Set $\mathrm{y} = \mathrm{A}\mathrm{x}^\star +$ noise with 10dB SNR
4. Tune λ and M to (hopefully) recover $\mathrm{x}^\star$ by solving (P)



Some numerical results

Synthetic setups :

1. Sample a synthetic $\mathrm{A} \in \mathbb{R}^{m \times n}$ with $(m, n) = (500, 1000)$
2. Generate a k-sparse vector $\mathrm{x}^\star$ with $k \in \{5, \ldots, 20\}$
3. Set $\mathrm{y} = \mathrm{A}\mathrm{x}^\star +$ noise with 10dB SNR
4. Tune λ and M to (hopefully) recover $\mathrm{x}^\star$ by solving (P)