Discrete Optimization Methods for L0-Penalized Sparse Problems

Theo Guyard with Cedric Herzet, Clement Elvira and Ayse-Nur Arslan
Inria and INSA, Rennes, France
theo.guyard@insa-rennes.fr

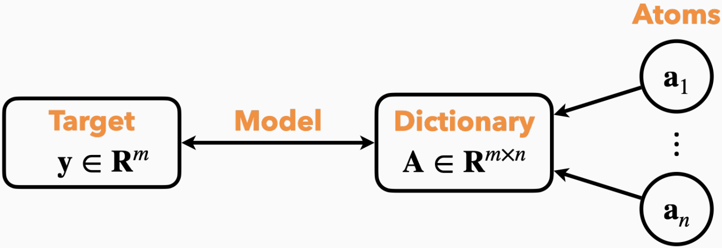
PGMO days | 29–30 November 2022

Sparse problems

Sparse problems

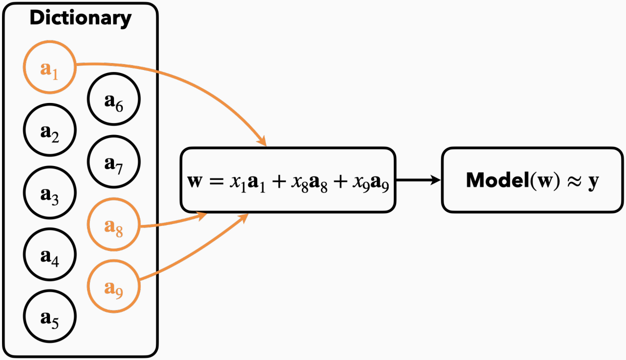
Sparse model components

- Target $y \in \mathbb{R}^m$ to be decomposed
- Atoms $a_i \in \mathbb{R}^m$ as basic components
- Dictionary $A \in \mathbb{R}^{m \times n}$ gathering the atoms
- Model linking y and A



Sparse problems

Objective : Fit the target with a sparse combination of atoms



Sparse problems

Rough formulation

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

Sparse problems

Rough formulation

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

Remark : Entries in x *weight each atom in the dictionary.*

Sparse problems

Rough formulation

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

Remark : Entries in x *weight each atom in the dictionary.*

$\rightsquigarrow$

ℓ_0-penalized problem

$$p^\star = \min F(\mathrm{y}, \mathrm{Ax}) + \lambda \|\mathrm{x}\|_0 \qquad (P)$$

Sparse problems

Rough formulation

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

Remark : Entries in x *weight each atom in the dictionary.*

$$\rightsquigarrow$$

ℓ_0-penalized problem

$$p^\star = \min F(\mathrm{y}, \mathrm{A}\mathrm{x}) + \lambda\|\mathrm{x}\|_0 \qquad (P)$$

- F ensures the model fitting
- $\|\cdot\|_0$ enforces sparsity
- $\lambda > 0$ controls the trade-off

Sparse problems

Rough formulation

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

Remark : Entries in x *weight each atom in the dictionary.*

$\rightsquigarrow$

ℓ_0-penalized problem

$$p^\star = \min F(\mathrm{y}, \mathrm{Ax}) + \lambda \|\mathrm{x}\|_0 \qquad (P)$$

- F ensures the model fitting
- $\|\cdot\|_0$ enforces sparsity
- $\lambda > 0$ controls the trade-off

(P) is NP-hard !

Mixed-Integer Program (MIP) formulation

MIP formulations

MIP

- Optimization problem
- Objective and constraints
- Continuous and integer variables

MIP formulations

MIP

- Optimization problem
- Objective and constraints
- Continuous and integer variables

Solver friendly structures

- Linear and quadratic expressions (GLPK, Cbc, CPLEX, Gurobi, ...)
- Conic expressions (Knitro, Mosek, ...)
- Non-linear expressions (Couenne, ...)

MIP formulations

MIP

- Optimization problem
- Objective and constraints
- Continuous and integer variables

Solver friendly structures

- Linear and quadratic expressions (GLPK, Cbc, CPLEX, Gurobi, ...)
- Conic expressions (Knitro, Mosek, ...)
- Non-linear expressions (Couenne, ...)

The ℓ_0-norm is not solver friendly !

Specially-Ordered Set (SOS) approach

SOS approach

Idea : Model the nullity in x using a binary variable.

$$\|x\|_0 = \begin{cases} 0 & \text{if } x = 0 \\ 1 & \text{if } x \neq 0 \end{cases}$$

SOS approach

Idea : Model the nullity in x using a binary variable.

$$\|x\|_0 = \begin{cases} 0 & \text{if } x = 0 \\ 1 & \text{if } x \neq 0 \end{cases}$$

Modelling nullity in x

Let $z \in \{0, 1\}$, then the constraint

$$x(1 - z) = 0 \iff (x, z) \in SOS_1$$

models the nullity in x entries.

SOS approach

Idea : Model the nullity in x using a binary variable.

$$\|x\|_0 = \begin{cases} 0 & \text{if } x = 0 \\ 1 & \text{if } x \neq 0 \end{cases}$$

Modelling nullity in x

Let $\mathrm{z} \in \{0, 1\}$, then the constraint

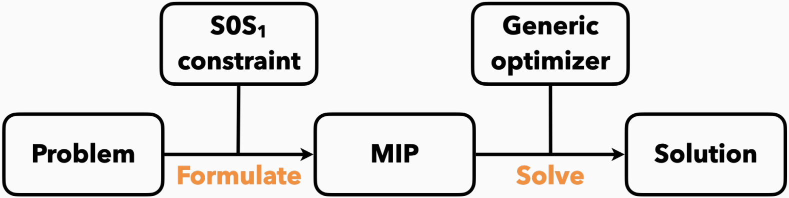
$$\mathrm{x}(1 - \mathrm{z}) = 0 \iff (\mathrm{x}, \mathrm{z}) \in SOS_1$$

models the nullity in x entries.

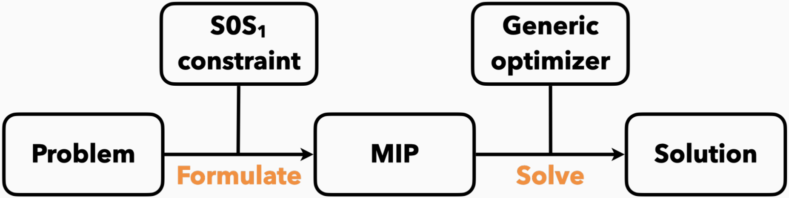
Solver friendly formulation

$$p^\star = \begin{cases} \min & F(\mathrm{y}, \mathrm{Ax}) + \lambda \mathbf{1}^\top \mathrm{z} \\ \text{s.t.} & (\mathrm{x}, \mathrm{z}) \in SOS_1 \\ & \mathrm{x} \in \mathbb{R}^n, \ \mathrm{z} \in \{0, 1\}^n \end{cases} \qquad (P_{sos})$$

SOS approach



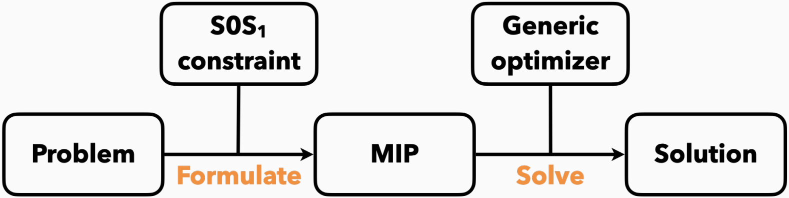
SOS approach



Pros

- ✓ Standard formulation
- ✓ Explicitly models sparsity
- ✓ Simple

SOS approach



Pros

- ✓ Standard formulation
- ✓ Explicitly models sparsity
- ✓ Simple

Cons

- ✗ Twice more variables
- ✗ Handling $\mathtt{SOS}_1$ constraints
- ✗ New layer of complexity

Branch-and-Bound (BnB) approach

BnB approach

BnB algorithms

- Separation of the search space
- Enumerate all feasible solutions
- Use rules to discard irrelevant candidates

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

BnB approach

BnB algorithms

- Separation of the search space
- Enumerate all feasible solutions
- Use rules to discard irrelevant candidates

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

Particularization to (P)

- Node defined as $\nu = (\mathcal{S}_0, \mathcal{S}_1, \bar{\mathcal{S}})$
  - $\mathcal{S}_0$: indices of x fixed to zero
  - $\mathcal{S}_1$: indices of x fixed to non-zero
  - $\bar{\mathcal{S}}$: indices of x not fixed yet
- Decisions about the nullity in the problem variable

BnB approach

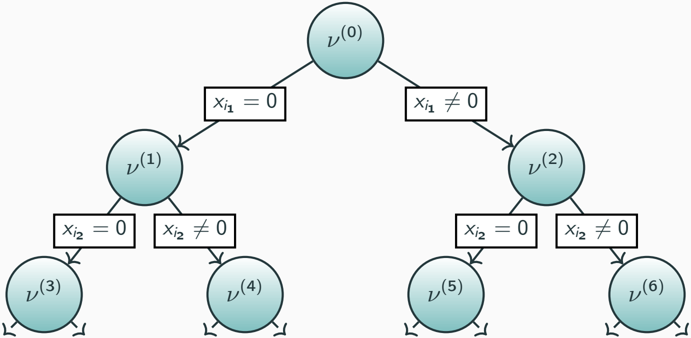


Figure 1: Example of BnB tree.

BnB approach

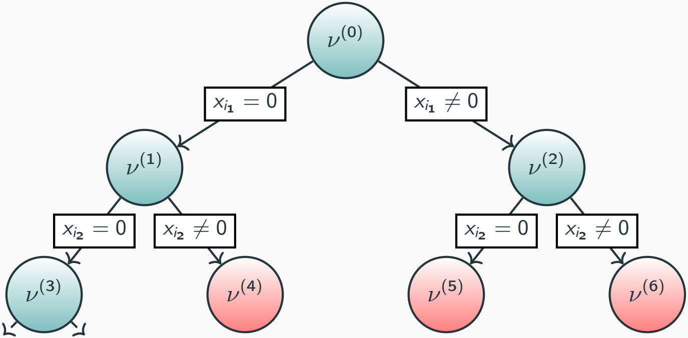


Figure 1: Example of BnB tree.

BnB approach

Node $\nu = (\mathcal{S}_0, \mathcal{S}_1, \bar{\mathcal{S}})$ **:** Does a solution of (P) match the constraints ?

BnB approach

Node $\nu = (\mathcal{S}_0, \mathcal{S}_1, \bar{\mathcal{S}})$ **:** Does a solution of (P) match the constraints ?

Problem at node ν

$$p^\nu = \begin{cases} \min & F(\mathbf{y}, \mathbf{A}\mathbf{x}) + \lambda \|\mathbf{x}\|_0 \\ \text{s.t.} & \mathbf{x}_{\mathcal{S}_0} = 0,\ \mathbf{x}_{\mathcal{S}_1} \neq 0 \end{cases} \tag{P^ν}$$

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

BnB approach

Node $\nu = (\mathcal{S}_0, \mathcal{S}_1, \bar{\mathcal{S}})$ **:** Does a solution of (P) match the constraints ?

Problem at node ν

$$p^\nu = \begin{cases} \min & F(y, Ax) + \lambda\|x\|_0 \\ \text{s.t.} & x_{\mathcal{S}_0} = 0,\ x_{\mathcal{S}_1} \neq 0 \end{cases} \qquad (P^\nu)$$

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

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

BnB approach

Node $\nu = (\mathcal{S}_0, \mathcal{S}_1, \bar{\mathcal{S}})$ **:** Does a solution of (P) match the constraints ?

Problem at node ν

$$p^\nu = \begin{cases} \min & F(\mathrm{y}, \mathrm{Ax}) + \lambda\|\mathrm{x}\|_0 \\ \text{s.t.} & \mathrm{x}_{\mathcal{S}_0} = 0,\ \mathrm{x}_{\mathcal{S}_1} \neq 0 \end{cases} \tag{P^ν}$$

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

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

BnB approach

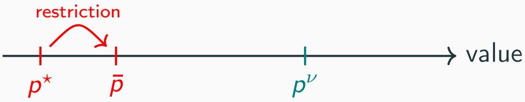
Node $\nu = (\mathcal{S}_0, \mathcal{S}_1, \bar{\mathcal{S}})$ **:** Does a solution of (P) match the constraints ?

Problem at node ν

$$p^\nu = \begin{cases} \min & F(\mathrm{y}, \mathrm{Ax}) + \lambda \|\mathrm{x}\|_0 \\ \text{s.t.} & \mathrm{x}_{\mathcal{S}_0} = 0,\ \mathrm{x}_{\mathcal{S}_1} \neq 0 \end{cases} \tag{P^ν}$$

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

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



BnB approach

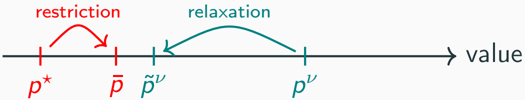
Node $\nu = (\mathcal{S}_0, \mathcal{S}_1, \bar{\mathcal{S}})$ **:** Does a solution of (P) match the constraints ?

Problem at node ν

$$p^\nu = \begin{cases} \min & F(\mathrm{y}, \mathrm{Ax}) + \lambda\|\mathrm{x}\|_0 \\ \text{s.t.} & \mathrm{x}_{\mathcal{S}_0} = 0,\ \mathrm{x}_{\mathcal{S}_1} \neq 0 \end{cases} \tag{P^ν}$$

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

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



BnB approach

Finding some upper bound $\bar{p} \geq p^\star$

- Keep track of the best objective value known for (P)
- Refine $\bar{p}$ using heuristics during the tree exploration

BnB approach

Finding some upper bound $\bar{p} \geq p^\star$

- Keep track of the best objective value known for (P)
- Refine $\bar{p}$ using heuristics during the tree exploration

Finding some lower bound $\tilde{p}^\nu \leq p^\nu$

- Relax the objective function of (P^ν) into a convex function

BnB approach

Relaxing the ℓ_0-norm at $\nu = (\mathcal{S}_0, \mathcal{S}_1, \bar{\mathcal{S}})$

BnB approach

Relaxing the ℓ_0-norm at $\nu = (\mathcal{S}_0, \mathcal{S}_1, \bar{\mathcal{S}})$

$\rightarrow$ Separable

BnB approach

Relaxing the ℓ_0-norm at $\nu = (\mathcal{S}_0, \mathcal{S}_1, \bar{\mathcal{S}})$

$\rightarrow$ Separable

$\rightarrow$ Closed-form expression for indices in $\mathcal{S}_0$ or $\mathcal{S}_1$

BnB approach

Relaxing the ℓ_0-norm at $\nu = (\mathcal{S}_0, \mathcal{S}_1, \bar{\mathcal{S}})$

$\rightarrow$ Separable

$\rightarrow$ Closed-form expression for indices in $\mathcal{S}_0$ or $\mathcal{S}_1$

$$\|\mathbf{x}\|_0 = \underbrace{\|\mathbf{x}_{\mathcal{S}_0}\|_0}_{0} + \underbrace{\|\mathbf{x}_{\mathcal{S}_1}\|_0}_{|\mathcal{S}_1|} + \|\mathbf{x}_{\bar{\mathcal{S}}}\|_0$$

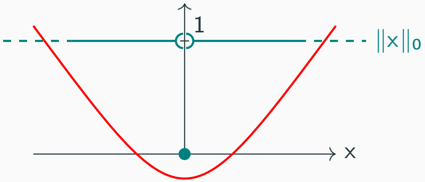
BnB approach

Relaxing the ℓ_0-norm at $\nu = (\mathcal{S}_0, \mathcal{S}_1, \bar{\mathcal{S}})$

→ Separable

→ Closed-form expression for indices in $\mathcal{S}_0$ or $\mathcal{S}_1$

$$\|x\|_0 = \underbrace{\|x_{\mathcal{S}_0}\|_0}_{0} + \underbrace{\|x_{\mathcal{S}_1}\|_0}_{|\mathcal{S}_1|} + \|x_{\bar{\mathcal{S}}}\|_0$$



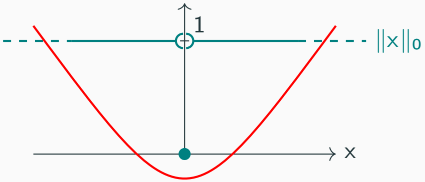
BnB approach

Relaxing the ℓ_0-norm at $\nu = (\mathcal{S}_0, \mathcal{S}_1, \bar{\mathcal{S}})$

$\rightarrow$ Separable

$\rightarrow$ Closed-form expression for indices in $\mathcal{S}_0$ or $\mathcal{S}_1$

$$\|\mathrm{x}\|_0 = \underbrace{\|\mathrm{x}_{\mathcal{S}_0}\|_0}_{0} + \underbrace{\|\mathrm{x}_{\mathcal{S}_1}\|_0}_{|\mathcal{S}_1|} + \|\mathrm{x}_{\bar{\mathcal{S}}}\|_0$$



No valid convex relaxation !

BnB approach

Perturbated ℓ_0-penalized problem

Rather consider

$$p^\star = \min F(y, Ax) + \lambda(\|x\|_0 + G(x)) \qquad (P_G)$$

where $G(\cdot)$ is a convex function.

BnB approach

Perturbated ℓ_0-penalized problem

Rather consider

$$p^\star = \min F(\mathrm{y}, \mathrm{Ax}) + \lambda(\|\mathrm{x}\|_0 + G(\mathrm{x})) \qquad (P_G)$$

where $G(\cdot)$ is a convex function.

Case $G(\mathrm{x}) = \mathrm{Ind}(\|\mathrm{x}\|_\infty \leq M)$

BnB approach

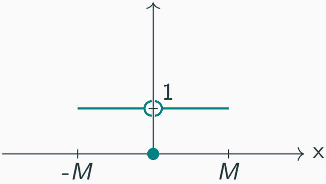
Perturbated ℓ_0-penalized problem

Rather consider

$$p^\star = \min F(\mathrm{y}, \mathrm{Ax}) + \lambda(\|\mathrm{x}\|_0 + G(\mathrm{x})) \qquad (P_G)$$

where $G(\cdot)$ is a convex function.

Case $G(\mathrm{x}) = \mathrm{Ind}(\|\mathrm{x}\|_\infty \leq M)$



BnB approach

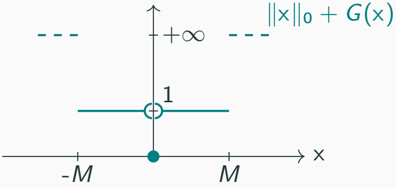
Perturbated ℓ_0-penalized problem

Rather consider

$$p^\star = \min F(\mathrm{y}, \mathrm{Ax}) + \lambda(\|\mathrm{x}\|_0 + G(\mathrm{x})) \qquad (P_G)$$

where $G(\cdot)$ is a convex function.

Case $G(\mathrm{x}) = \mathrm{Ind}(\|\mathrm{x}\|_\infty \leq M)$



BnB approach

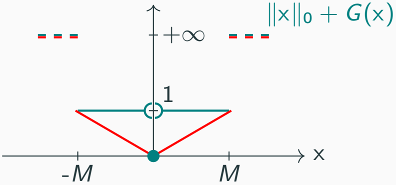
Perturbated ℓ_0-penalized problem

Rather consider

$$p^\star = \min F(\mathrm{y}, \mathrm{Ax}) + \lambda(\|\mathrm{x}\|_0 + G(\mathrm{x})) \qquad (P_G)$$

where $G(\cdot)$ is a convex function.

Case $G(\mathrm{x}) = \mathrm{Ind}(\|\mathrm{x}\|_\infty \leq M)$



BnB approach

Perturbated ℓ_0-penalized problem

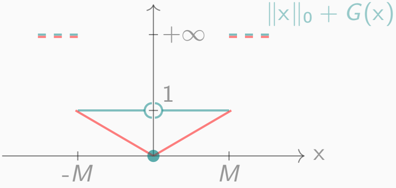
Rather consider

$$p^\star = \min F(\mathrm{y}, \mathrm{Ax}) + \lambda(\|\mathrm{x}\|_0 + G(\mathrm{x})) \qquad (P_G)$$

where $G(\cdot)$ is a convex function.

Case $G(\mathrm{x}) = \mathrm{Ind}(\|\mathrm{x}\|_\infty \leq M)$

Case $G(\mathrm{x}) = \alpha\|\mathrm{x}\|_2^2$



BnB approach

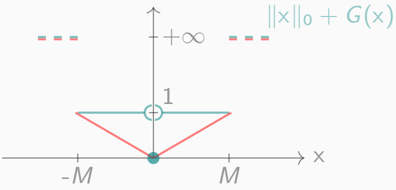
Perturbated ℓ_0-penalized problem

Rather consider

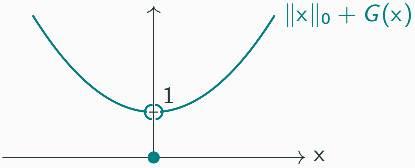
$$p^\star = \min F(\mathrm{y}, \mathrm{Ax}) + \lambda(\|\mathrm{x}\|_0 + G(\mathrm{x})) \qquad (P_G)$$

where $G(\cdot)$ is a convex function.

Case $G(\mathrm{x}) = \mathrm{Ind}(\|\mathrm{x}\|_\infty \leq M)$



Case $G(\mathrm{x}) = \alpha\|\mathrm{x}\|_2^2$



BnB approach

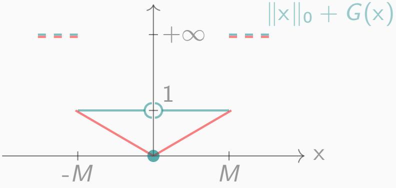
Perturbated ℓ_0-penalized problem

Rather consider

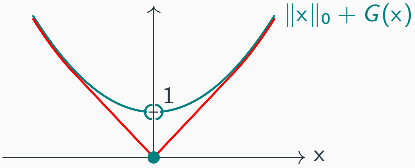
$$p^\star = \min F(\mathrm{y}, \mathrm{Ax}) + \lambda(\|\mathrm{x}\|_0 + G(\mathrm{x})) \qquad (P_G)$$

where $G(\cdot)$ is a convex function.

Case $G(\mathrm{x}) = \mathrm{Ind}(\|\mathrm{x}\|_\infty \leq M)$



Case $G(\mathrm{x}) = \alpha\|\mathrm{x}\|_2^2$



BnB approach

Explore less and faster

BnB approach

Explore less and faster

- Coordinate Descent to solve the relaxations *(Hazimeh et al., 2021)*

BnB approach

Explore less and faster

- Coordinate Descent to solve the relaxations *(Hazimeh et al., 2021)*
- Screening to accelerate the CD *(El Ghaoui et al., 2011)*

BnB approach

Explore less and faster

- Coordinate Descent to solve the relaxations *(Hazimeh et al., 2021)*
- Screening to accelerate the CD *(El Ghaoui et al., 2011)*
- Node-screening to discard uninteresting nodes *(Guyard et al., 2022)*

BnB approach

Explore less and faster

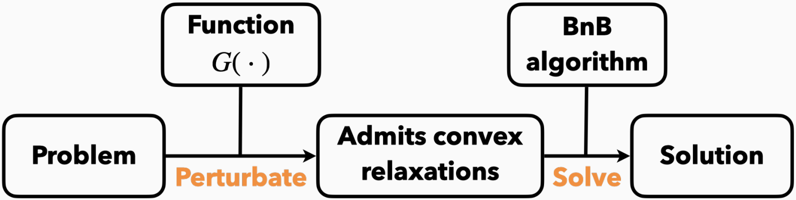
- Coordinate Descent to solve the relaxations *(Hazimeh et al., 2021)*
- Screening to accelerate the CD *(El Ghaoui et al., 2011)*
- Node-screening to discard uninteresting nodes *(Guyard et al., 2022)*
- Early pruning to save computations *(Samain et al., 2022)*

BnB approach

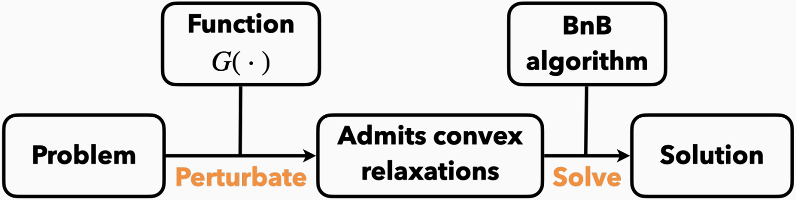
Explore less and faster

- Coordinate Descent to solve the relaxations *(Hazimeh et al., 2021)*
- Screening to accelerate the CD *(El Ghaoui et al., 2011)*
- Node-screening to discard uninteresting nodes *(Guyard et al., 2022)*
- Early pruning to save computations *(Samain et al., 2022)*
- ...

BnB approach



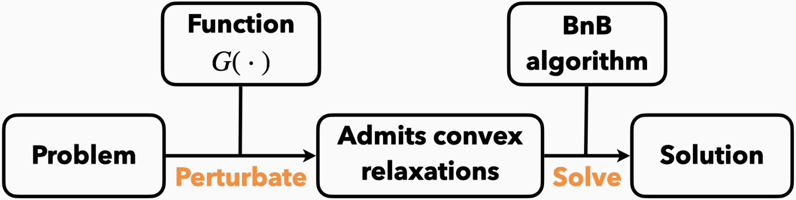
BnB approach



Pros

- ✓ Tailored BnB algorithm
- ✓ Exploit sparsity
- ✓ Faster convergence

BnB approach



Pros

- ✓ Tailored BnB algorithm
- ✓ Exploit sparsity
- ✓ Faster convergence

Cons

- ✗ Perturb the problem
- ✗ New hyperparameters to tune

Conclusion

Conclusion

Addressing ℓ_0-penalized problems

Conclusion

Addressing ℓ_0-penalized problems

- Leveraging $\texttt{SOS}_1$ constraints
  - ✓ MIP formulation
  - ✓ Generic solvers
  - ✗ Still hard to handle

Conclusion

Addressing ℓ_0-penalized problems

- Leveraging $\texttt{SOS}_1$ constraints
  - ✓ MIP formulation
  - ✓ Generic solvers
  - ✗ Still hard to handle

- Using a Branch-and-Bound procedure
  - ✓ Specialized solution method
  - ✓ Efficient to solve the problem
  - ✗ Have to perturb the problem

Advertizing time

https://github.com/TheoGuyard/El0ps.jl