A Unified Framework for Mixed-Integer Optimization: Nonlinear Formulations and Scalable Algorithms

Ryan Cory-Wright

August 2019

ORC, Massachusetts Institute of Technology
Joint work with Dimitris Bertsimas and Jean Pauphilet
Preprint available at: ryancorywright.github.io

Motivation: Best Subset Selection

In high-dimensional regression, desirable to fit a parsimonious model which uses at most k features. Achieved via best subset selection

$$\min_{\boldsymbol{x}\in\mathbb{R}^p} \quad \|\boldsymbol{y}-\boldsymbol{X}\boldsymbol{x}\|_2^2 \text{ s.t.} \quad \|\boldsymbol{x}\|_0 \le k.$$

Motivation: Best Subset Selection

In high-dimensional regression, desirable to fit a parsimonious model which uses at most k features. Achieved via best subset selection

$$\min_{\boldsymbol{x}\in\mathbb{R}^p} \quad \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{x}\|_2^2 \text{ s.t.} \quad \|\boldsymbol{x}\|_0 \leq k.$$

Introducing binary vector $\boldsymbol{z}$, problem is a MIP:

$$\min_{\boldsymbol{x}\in\mathbb{R}^p, \boldsymbol{z}\in\{0,1\}^p} \quad \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{x}\|_2^2 \text{ s.t.} \quad \sum_i z_i \leq k, x_i = 0 \text{ if } z_i = 0, \forall i.$$

Motivation: Best Subset Selection

In high-dimensional regression, desirable to fit a parsimonious model which uses at most k features. Achieved via best subset selection

$$\min_{\boldsymbol{x} \in \mathbb{R}^p} \quad \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{x}\|_2^2 \text{ s.t. } \quad \|\boldsymbol{x}\|_0 \leq k.$$

Introducing binary vector $\boldsymbol{z}$, problem is a MIP:

$$\min_{\boldsymbol{x} \in \mathbb{R}^p, \boldsymbol{z} \in \{0,1\}^p} \quad \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{x}\|_2^2 \text{ s.t. } \quad \sum_i z_i \leq k, x_i = 0 \text{ if } z_i = 0, \forall i.$$

A problem with logical structure: $x = 0$ *if* $z = 0$ for z binary.

A Family of Problems With Logical Constraints

Logical on/off structure appears in many important problems!

Central problems in optimization/statistics have logical relations between continuous variables $\boldsymbol{x}$, binary variables $\boldsymbol{z}$: $\boldsymbol{x} = \mathbf{0}$ *if* $\boldsymbol{z} = \mathbf{0}$.

A Family of Problems With Logical Constraints

Logical on/off structure appears in many important problems!

Central problems in optimization/statistics have logical relations between continuous variables $\boldsymbol{x}$, binary variables $\boldsymbol{z}$: $\boldsymbol{x} = \mathbf{0}$ *if* $\boldsymbol{z} = \mathbf{0}$.

- **Best subset selection**: $\min_{\boldsymbol{x}} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{x}\|_2^2 : \ \|\boldsymbol{x}\|_0 \leq k$.

A Family of Problems With Logical Constraints

Logical on/off structure appears in many important problems!

Central problems in optimization/statistics have logical relations between continuous variables $\boldsymbol{x}$, binary variables $\boldsymbol{z}$: $\boldsymbol{x} = \mathbf{0}$ *if* $\boldsymbol{z} = \mathbf{0}$.

- **Best subset selection**: $\min_{\boldsymbol{x}} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{x}\|_2^2 : \ \|\boldsymbol{x}\|_0 \leq k$.
- **Compressed sensing**: $\min_{\boldsymbol{x}} \|\boldsymbol{x}\|_0 : \ \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$.

A Family of Problems With Logical Constraints

Logical on/off structure appears in many important problems!

Central problems in optimization/statistics have logical relations between continuous variables $\boldsymbol{x}$, binary variables $\boldsymbol{z}$: $\boldsymbol{x} = \mathbf{0}$ *if* $\boldsymbol{z} = \mathbf{0}$.

- **Best subset selection**: $\min_{\boldsymbol{x}} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{x}\|_2^2 : \ \|\boldsymbol{x}\|_0 \leq k$.
- **Compressed sensing**: $\min_{\boldsymbol{x}} \|\boldsymbol{x}\|_0 : \ \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$.
- **Facility Location**:

$$\begin{aligned} &\min_{\boldsymbol{z}\in\{0,1\}^n} \min_{\boldsymbol{X}\in\mathbb{R}_+^{n\times m}} \boldsymbol{c}^\top \boldsymbol{z} + \sum_{j=1}^m \sum_{i=1}^n C_{ij} X_{ij} \\ \text{s.t. } &\sum_{j=1}^m X_{ij} \leq U_i, \ \forall i \in [n], \ \sum_{i=1}^n X_{ij} = d_j, \ \forall j \in [m], \\ &X_{ij} = 0 \ \text{ if } \ z_i = 0, \ \forall i \in [n], j \in [m]. \end{aligned} \tag{1}$$

A Family of Problems With Logical Constraints

Logical on/off structure appears in many important problems!

Central problems in optimization/statistics have logical relations between continuous variables $\boldsymbol{x}$, binary variables $\boldsymbol{z}$: $\boldsymbol{x} = \mathbf{0}$ *if* $\boldsymbol{z} = \mathbf{0}$.

- **Best subset selection**: $\min_{\boldsymbol{x}} \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{x}\|_2^2 : \ \|\boldsymbol{x}\|_0 \leq k$.
- **Compressed sensing**: $\min_{\boldsymbol{x}} \|\boldsymbol{x}\|_0 : \ \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$.
- **Facility Location**:

$$\begin{aligned} \min_{\boldsymbol{z} \in \{0,1\}^n} \min_{\boldsymbol{X} \in \mathbb{R}_+^{n \times m}} \ & \boldsymbol{c}^\top \boldsymbol{z} + \sum_{j=1}^m \sum_{i=1}^n C_{ij} X_{ij} \\ \text{s.t.} \ \sum_{j=1}^m X_{ij} \leq U_i, \ \forall i \in [n], \ & \sum_{i=1}^n X_{ij} = d_j, \ \forall j \in [m], \\ & X_{ij} = 0 \ \text{ if } \ z_i = 0, \ \forall i \in [n], j \in [m]. \end{aligned} \tag{1}$$

- Many others: **Sparse Portfolio Selection, Network Design, Unit Commitment, Scheduling**.

Logical Constraints: The State of The Art

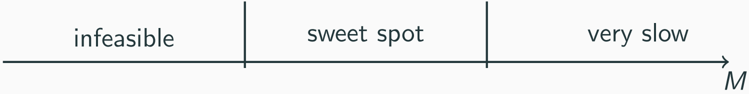
Optimization textbooks propose an exact approach to on/off constraints:

- Big-M approach: rewrite "$x = 0$ if $z = 0$" as $-Mz \leq x \leq Mz$.
- Finding the right M is hard!

Logical Constraints: The State of The Art

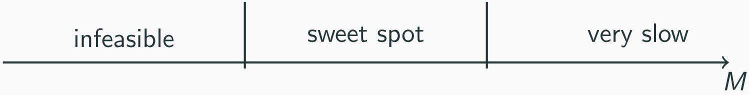
Optimization textbooks propose an exact approach to on/off constraints:

- Big-M approach: rewrite "$x = 0$ if $z = 0$" as $-Mz \leq x \leq Mz$.
- Finding the right M is hard!



Logical Constraints: The State of The Art

Optimization textbooks propose an exact approach to on/off constraints:

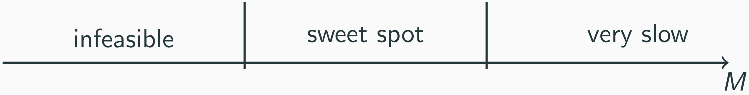
- Big-M approach: rewrite "$x = 0$ if $z = 0$" as $-Mz \leq x \leq Mz$.
- Finding the right M is hard!
- 

Statistics textbooks suggest using heuristics:

- Convex optimization methods such as Lasso.
- Sometimes work well, but no optimality guarantees.

Logical Constraints: The State of The Art

Optimization textbooks propose an exact approach to on/off constraints:

- Big-M approach: rewrite "$x = 0$ if $z = 0$" as $-Mz \leq x \leq Mz$.
- Finding the right M is hard!
- 

Statistics textbooks suggest using heuristics:

- Convex optimization methods such as Lasso.
- Sometimes work well, but no optimality guarantees.

State-of-the-art methods are either heuristics or too slow.
Alternatives are needed.

Modelling Contributions

- We propose a **non-linear reformulation** of the logical constraints, substituting xz for x.

Modelling Contributions

- We propose a **non-linear reformulation** of the logical constraints, substituting xz for x.
- We show that adding a $\frac{1}{2\gamma}\|\boldsymbol{x}\|_2^2$ **ridge regularizer** to the objective is a viable and often more scalable alternative to big-M.

Algorithmic Contributions

- By using strong duality, we derive a saddle-point reformulation, which is **exactly** solvable via an outer-approximation procedure.

Algorithmic Contributions

- By using strong duality, we derive a saddle-point reformulation, which is **exactly** solvable via an outer-approximation procedure.
- We obtain provably **near-optimal solutions in polynomial time** by solving a Boolean relaxation efficiently.

Algorithmic Contributions

- By using strong duality, we derive a saddle-point reformulation, which is **exactly** solvable via an outer-approximation procedure.
- We obtain provably **near-optimal solutions in polynomial time** by solving a Boolean relaxation efficiently.
- Our approach **is scalable**: it solves sparse regression problems with $100,000$s of covariates, sparse portfolio selection problems with 1000s of securities, network design problems with 100s of nodes.

The Unified Framework

A Mixed-Integer Nonlinear Program With Logical Constraints

$$\min_{\boldsymbol{z}\in\mathcal{Z},\ \boldsymbol{x}\in\mathbb{R}^n} \boldsymbol{c}^\top \boldsymbol{z} + \underbrace{g(\boldsymbol{x})}_{\text{convex function}} + \underbrace{\Omega(\boldsymbol{x})}_{\text{regularizer}} \quad \text{s.t. } \underbrace{x_i = 0 \text{ if } z_i = 0}_{\text{logical constraint}},\ \forall i,$$

where:

A Mixed-Integer Nonlinear Program With Logical Constraints

$$\min_{\boldsymbol{z}\in\mathcal{Z},\ \boldsymbol{x}\in\mathbb{R}^n} \ \boldsymbol{c}^\top \boldsymbol{z} + \underbrace{g(\boldsymbol{x})}_{\text{convex function}} + \underbrace{\Omega(\boldsymbol{x})}_{\text{regularizer}} \ \text{ s.t. } \underbrace{x_i = 0 \text{ if } z_i = 0}_{\text{logical constraint}}, \ \forall i,$$

where:

- $\mathcal{Z} \subseteq \{0,1\}^n$ constrains $\boldsymbol{z}$, e.g., cardinality constraint $\boldsymbol{e}^\top \boldsymbol{z} \leq k$.

A Mixed-Integer Nonlinear Program With Logical Constraints

$$\min_{\boldsymbol{z}\in\mathcal{Z},\ \boldsymbol{x}\in\mathbb{R}^n} \boldsymbol{c}^\top \boldsymbol{z} + \underbrace{g(\boldsymbol{x})}_{\text{convex function}} + \underbrace{\Omega(\boldsymbol{x})}_{\text{regularizer}} \text{ s.t. } \underbrace{x_i = 0 \text{ if } z_i = 0}_{\text{logical constraint}},\ \forall i,$$

where:

- $\mathcal{Z} \subseteq \{0,1\}^n$ constrains $\boldsymbol{z}$, e.g., cardinality constraint $\boldsymbol{e}^\top \boldsymbol{z} \leq k$.
- We model convex constraints $\boldsymbol{x} \in \mathcal{X}$ via $g(\boldsymbol{x}) = +\infty$ if $\boldsymbol{x} \notin \mathcal{X}$.
- The regularizer $\Omega(\cdot)$ convexifies the logical constraints.

A Mixed-Integer Nonlinear Program With Logical Constraints

$$\min_{\boldsymbol{z}\in\mathcal{Z},\ \boldsymbol{x}\in\mathbb{R}^n} \boldsymbol{c}^\top \boldsymbol{z} + \underbrace{g(\boldsymbol{x})}_{\text{convex function}} + \underbrace{\Omega(\boldsymbol{x})}_{\text{regularizer}} \ \text{s.t.}\ \underbrace{x_i = 0 \text{ if } z_i = 0}_{\text{logical constraint}},\ \forall i,$$

where:

- $\mathcal{Z} \subseteq \{0,1\}^n$ constrains $\boldsymbol{z}$, e.g., cardinality constraint $\boldsymbol{e}^\top \boldsymbol{z} \le k$.
- We model convex constraints $\boldsymbol{x} \in \mathcal{X}$ via $g(\boldsymbol{x}) = +\infty$ if $\boldsymbol{x} \notin \mathcal{X}$.
- The regularizer $\Omega(\cdot)$ convexifies the logical constraints. It is one of:
    1. A big-M penalty: $\Omega(\boldsymbol{x}) = 0$ if $\|\boldsymbol{x}\|_\infty \le M$ and $+\infty$ otherwise.
    2. A ridge penalty: $\Omega(\boldsymbol{x}) = \frac{1}{2\gamma}\|\boldsymbol{x}\|_2^2$.

A Mixed-Integer Nonlinear Program With Logical Constraints

$$\min_{\boldsymbol{z}\in\mathcal{Z},\ \boldsymbol{x}\in\mathbb{R}^n} \boldsymbol{c}^\top \boldsymbol{z} + \underbrace{g(\boldsymbol{x})}_{\text{convex function}} + \underbrace{\Omega(\boldsymbol{x})}_{\text{regularizer}} \text{ s.t. } \underbrace{x_i = 0 \text{ if } z_i = 0}_{\text{logical constraint}},\ \forall i,$$

where:

- $\mathcal{Z} \subseteq \{0,1\}^n$ constrains $\boldsymbol{z}$, e.g., cardinality constraint $\boldsymbol{e}^\top \boldsymbol{z} \leq k$.
- We model convex constraints $\boldsymbol{x} \in \mathcal{X}$ via $g(\boldsymbol{x}) = +\infty$ if $\boldsymbol{x} \notin \mathcal{X}$.
- The regularizer $\Omega(\cdot)$ convexifies the logical constraints. It is one of:
  1. A big-M penalty: $\Omega(\boldsymbol{x}) = 0$ if $\|\boldsymbol{x}\|_\infty \leq M$ and $+\infty$ otherwise.
  2. A ridge penalty: $\Omega(\boldsymbol{x}) = \frac{1}{2\gamma}\|\boldsymbol{x}\|_2^2$.

All six problems on the second slide fit into this framework!

A Mixed-Integer Nonlinear Program With Logical Constraints

$$\min_{\boldsymbol{z}\in\mathcal{Z},\ \boldsymbol{x}\in\mathbb{R}^n} \boldsymbol{c}^\top \boldsymbol{z} + \underbrace{g(\boldsymbol{x})}_{\text{convex function}} + \underbrace{\Omega(\boldsymbol{x})}_{\text{regularizer}} \ \text{s.t.}\ \underbrace{x_i = 0 \text{ if } z_i = 0}_{\text{logical constraint}},\ \forall i,$$

where:

- $\mathcal{Z} \subseteq \{0,1\}^n$ constrains $\boldsymbol{z}$, e.g., cardinality constraint $\boldsymbol{e}^\top \boldsymbol{z} \leq k$.
- We model convex constraints $\boldsymbol{x} \in \mathcal{X}$ via $g(\boldsymbol{x}) = +\infty$ if $\boldsymbol{x} \notin \mathcal{X}$.
- The regularizer $\Omega(\cdot)$ convexifies the logical constraints. It is one of:
  1. A big-M penalty: $\Omega(\boldsymbol{x}) = 0$ if $\|\boldsymbol{x}\|_\infty \leq M$ and $+\infty$ otherwise.
  2. A ridge penalty: $\Omega(\boldsymbol{x}) = \frac{1}{2\gamma}\|\boldsymbol{x}\|_2^2$.

All six problems on the second slide fit into this framework!

- Allows us to solve all six problems using the same piece of code.

Simplifying the Problem

We rewrite the problem as

$$\min_{\boldsymbol{z} \in \mathcal{Z}} \; f(\boldsymbol{z}),$$

where

$$f(\boldsymbol{z}) = \min_{\boldsymbol{x}} \; \boldsymbol{c}^\top \boldsymbol{z} + \underbrace{g(\boldsymbol{x})}_{\text{convex}} + \underbrace{\Omega(\boldsymbol{x})}_{\textit{regularizer}} \quad \text{s.t. } \underbrace{x_i = 0 \text{ if } z_i = 0}_{\text{logical constraint}}, \; \forall i,$$

Simplifying the Problem

We rewrite the problem as

$$\min_{\boldsymbol{z}\in\mathcal{Z}} \; f(\boldsymbol{z}),$$

where

$$f(\boldsymbol{z}) = \min_{\boldsymbol{x}} \; \boldsymbol{c}^\top \boldsymbol{z} + \underbrace{g(\boldsymbol{x})}_{\text{convex}} + \underbrace{\Omega(\boldsymbol{x})}_{\textit{regularizer}} \quad \text{s.t. } \underbrace{x_i = 0 \text{ if } z_i = 0}_{\text{logical constraint}}, \; \forall i,$$

$$= \min_{\boldsymbol{x}} \; \boldsymbol{c}^\top \boldsymbol{z} + g(\boldsymbol{z} \circ \boldsymbol{x}) + \Omega(\boldsymbol{z} \circ \boldsymbol{x}) \qquad \text{non-linear}$$

Simplifying the Problem

We rewrite the problem as

$$\min_{\boldsymbol{z} \in \mathcal{Z}} \; f(\boldsymbol{z}),$$

where

$$\begin{aligned}
f(\boldsymbol{z}) &= \min_{\boldsymbol{x}} \; \boldsymbol{c}^\top \boldsymbol{z} + \underbrace{g(\boldsymbol{x})}_{\text{convex}} + \underbrace{\Omega(\boldsymbol{x})}_{\textit{regularizer}} \quad \text{s.t. } \underbrace{x_i = 0 \text{ if } z_i = 0}_{\text{logical constraint}}, \; \forall i, \\
&= \min_{\boldsymbol{x}} \; \boldsymbol{c}^\top \boldsymbol{z} + g(\boldsymbol{z} \circ \boldsymbol{x}) + \Omega(\boldsymbol{z} \circ \boldsymbol{x}) && \text{non-linear} \\
&= \max_{\boldsymbol{\alpha}} \; \boldsymbol{c}^\top \boldsymbol{z} + \underbrace{h(\boldsymbol{\alpha})}_{\text{concave}} - \sum_i z_i \Omega^\star(\alpha_i) && \text{strong duality}^1
\end{aligned}$$

---

[1] Assuming that $\forall \boldsymbol{z} \in \mathcal{Z}$, either the problem generated by $f(\boldsymbol{z})$ is infeasible or strong duality holds. True for all six problems on slide 2, and generally holds under a CQ.

Simplifying the Problem

We rewrite the problem as

$$\min_{\boldsymbol{z}\in\mathcal{Z}} \ f(\boldsymbol{z}),$$

where

$$\begin{aligned}
f(\boldsymbol{z}) &= \min_{\boldsymbol{x}} \ \boldsymbol{c}^\top \boldsymbol{z} + \underbrace{g(\boldsymbol{x})}_{\text{convex}} + \underbrace{\Omega(\boldsymbol{x})}_{\textit{regularizer}} \quad \text{s.t. } \underbrace{x_i = 0 \text{ if } z_i = 0}_{\text{logical constraint}}, \ \forall i, \\
&= \min_{\boldsymbol{x}} \ \boldsymbol{c}^\top \boldsymbol{z} + g(\boldsymbol{z}\circ\boldsymbol{x}) + \Omega(\boldsymbol{z}\circ\boldsymbol{x}) && \text{non-linear} \\
&= \max_{\boldsymbol{\alpha}} \ \boldsymbol{c}^\top \boldsymbol{z} + \underbrace{h(\boldsymbol{\alpha})}_{\text{concave}} - \sum_i z_i \Omega^\star(\alpha_i) && \text{strong duality}^1
\end{aligned}$$

which proves $f(z)$ is convex!

---

[1] Assuming that $\forall \boldsymbol{z} \in \mathcal{Z}$, either the problem generated by $f(\boldsymbol{z})$ is infeasible or strong duality holds. True for all six problems on slide 2, and generally holds under a CQ.

The Saddle-Point Reformulation

$$f(\boldsymbol{z}) = \max_{\boldsymbol{\alpha}} \ \boldsymbol{c}^\top \boldsymbol{z} + \underbrace{h(\boldsymbol{\alpha})}_{\text{concave}} - \sum_i z_i \Omega^\star(\alpha_i),$$

where

- $h(\cdot)$ is concave; the Fenchel conjugate of $g(\cdot)$ (up to a minus sign).
- $\Omega^\star(\alpha) = M|\alpha|$ for the big-M penalty.
- $\Omega^\star(\alpha) = \gamma/2\alpha^2$ for the ridge penalty.

So What?

The Outer-Approximation Method

Our saddle-point representation:

$$\min_{\boldsymbol{z}\in\mathcal{Z}} f(\boldsymbol{z}) = \min_{\boldsymbol{z}\in\mathcal{Z}} \max_{\boldsymbol{\alpha}} \ \boldsymbol{c}^\top \boldsymbol{z} + h(\boldsymbol{\alpha}) - \sum_i z_i \Omega^\star(\alpha_i)$$

lends itself to a tractable outer-approximation method.

The Outer-Approximation Method

Our saddle-point representation:

$$\min_{\boldsymbol{z} \in \mathcal{Z}} f(\boldsymbol{z}) = \min_{\boldsymbol{z} \in \mathcal{Z}} \max_{\boldsymbol{\alpha}} \boldsymbol{c}^\top \boldsymbol{z} + h(\boldsymbol{\alpha}) - \sum_i z_i \Omega^\star(\alpha_i)$$

lends itself to a tractable outer-approximation method.

- Fix $\boldsymbol{z}_0$, solve an easy convex program to obtain $\boldsymbol{\alpha}^\star(\boldsymbol{z}_0)$ and $f(\boldsymbol{z}_0)$.

The Outer-Approximation Method

Our saddle-point representation:

$$\min_{\boldsymbol{z} \in \mathcal{Z}} f(\boldsymbol{z}) = \min_{\boldsymbol{z} \in \mathcal{Z}} \max_{\boldsymbol{\alpha}} \boldsymbol{c}^\top \boldsymbol{z} + h(\boldsymbol{\alpha}) - \sum_i z_i \Omega^\star(\alpha_i)$$

lends itself to a tractable outer-approximation method.

- Fix $\boldsymbol{z}_0$, solve an easy convex program to obtain $\boldsymbol{\alpha}^\star(\boldsymbol{z}_0)$ and $f(\boldsymbol{z}_0)$.
- Obtain subgradient for all n indices **even those where** $z_{i,0} = 0$.

The Outer-Approximation Method

Our saddle-point representation:

$$\min_{\boldsymbol{z}\in\mathcal{Z}} f(\boldsymbol{z}) = \min_{\boldsymbol{z}\in\mathcal{Z}} \max_{\boldsymbol{\alpha}} \; \boldsymbol{c}^\top \boldsymbol{z} + h(\boldsymbol{\alpha}) - \sum_i z_i \Omega^\star(\alpha_i)$$

lends itself to a tractable outer-approximation method.

- Fix $\boldsymbol{z}_0$, solve an easy convex program to obtain $\boldsymbol{\alpha}^\star(\boldsymbol{z}_0)$ and $f(\boldsymbol{z}_0)$.
- Obtain subgradient for all n indices **even those where** $z_{i,0} = 0$.
- $f(\boldsymbol{z}) \geq f(\boldsymbol{z}_0) + \nabla f(\boldsymbol{z}_0)^\top(\boldsymbol{z} - \boldsymbol{z}_0)$ is a valid outer-approximation cut.

The Outer-Approximation Method

Our saddle-point representation:

$$\min_{\boldsymbol{z}\in\mathcal{Z}} f(\boldsymbol{z}) = \min_{\boldsymbol{z}\in\mathcal{Z}} \max_{\boldsymbol{\alpha}} \boldsymbol{c}^\top \boldsymbol{z} + h(\boldsymbol{\alpha}) - \sum_i z_i \Omega^\star(\alpha_i)$$

lends itself to a tractable outer-approximation method.

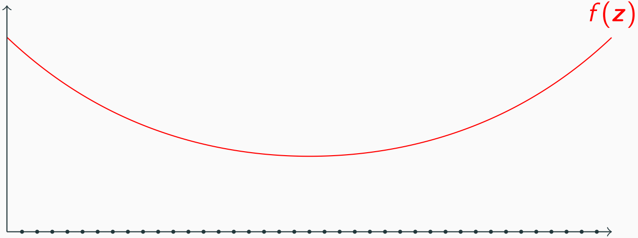
- Fix $\boldsymbol{z}_0$, solve an easy convex program to obtain $\boldsymbol{\alpha}^\star(\boldsymbol{z}_0)$ and $f(\boldsymbol{z}_0)$.
- Obtain subgradient for all n indices **even those where** $z_{i,0} = 0$.
- $f(\boldsymbol{z}) \geq f(\boldsymbol{z}_0) + \nabla f(\boldsymbol{z}_0)^\top (\boldsymbol{z} - \boldsymbol{z}_0)$ is a valid outer-approximation cut.
- Iteratively adding cuts, minimizing piecewise linear underestimator in Julia/CPLEX minimizes $f(\boldsymbol{z})$. Using Branch-and-Cut with lazy constraints solves entire problem using one branch-and-bound tree.
- As will see in numerical results, solves very large-scale problems.

The Outer Approximation Process

We solve the problem

$$\min_{\boldsymbol{z} \in \mathcal{Z}} f(\boldsymbol{z})$$

by iteratively minimizing a piecewise linear underestimator of f.

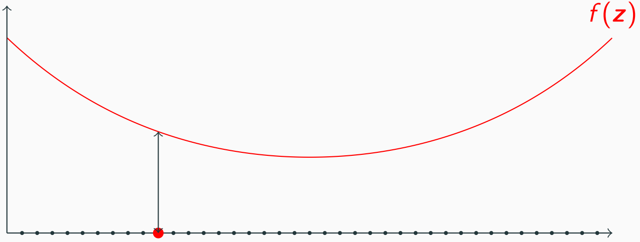


The Outer Approximation Process

We solve the problem

$$\min_{\boldsymbol{z} \in \mathcal{Z}} f(\boldsymbol{z})$$

by iteratively minimizing a piecewise linear underestimator of f.

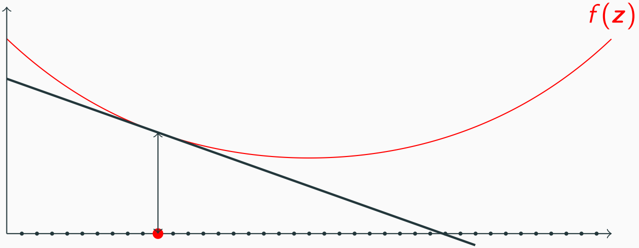


The Outer Approximation Process

We solve the problem

$$\min_{\boldsymbol{z} \in \mathcal{Z}} f(\boldsymbol{z})$$

by iteratively minimizing a piecewise linear underestimator of f.

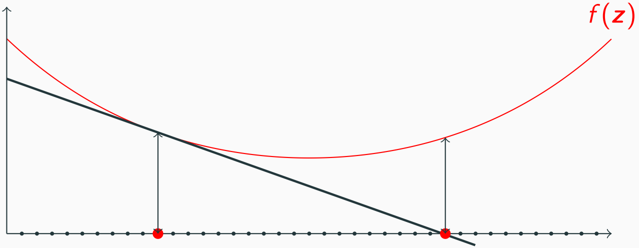


The Outer Approximation Process

We solve the problem

$$\min_{\boldsymbol{z} \in \mathcal{Z}} f(\boldsymbol{z})$$

by iteratively minimizing a piecewise linear underestimator of f.

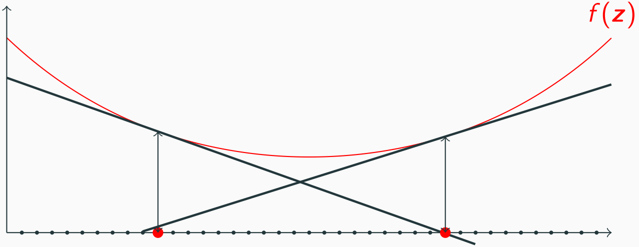


The Outer Approximation Process

We solve the problem

$$\min_{\boldsymbol{z} \in \mathcal{Z}} f(\boldsymbol{z})$$

by iteratively minimizing a piecewise linear underestimator of f.

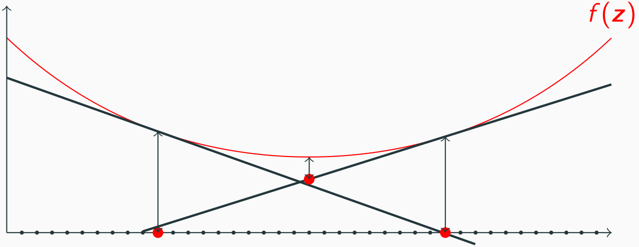


The Outer Approximation Process

We solve the problem

$$\min_{\boldsymbol{z} \in \mathcal{Z}} f(\boldsymbol{z})$$

by iteratively minimizing a piecewise linear underestimator of f.

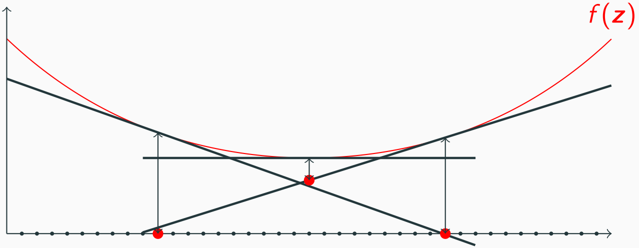


The Outer Approximation Process

We solve the problem

$$\min_{\boldsymbol{z} \in \mathcal{Z}} f(\boldsymbol{z})$$

by iteratively minimizing a piecewise linear underestimator of f.



A Boolean Relaxation

$$\min_{\boldsymbol{z} \in \mathrm{Conv}(\mathcal{Z})} \max_{\boldsymbol{\alpha}} \; \boldsymbol{c}^\top \boldsymbol{z} + h(\boldsymbol{\alpha}) - \sum_i z_i \Omega^\star(\alpha_i)$$

A Boolean Relaxation

$$\min_{\boldsymbol{z} \in \mathrm{Conv}(\mathcal{Z})} \max_{\boldsymbol{\alpha}} \ \boldsymbol{c}^\top \boldsymbol{z} + h(\boldsymbol{\alpha}) - \sum_i z_i \Omega^\star(\alpha_i)$$

- Solve by sub-gradient descent method, or transform to SOCP.
- Randomly rounding relaxation $\boldsymbol{z}^\star$ according to $z_i \sim \mathrm{Bernoulli}(z_i^\star)$ gives a Boolean vector $\boldsymbol{z}$. How good is it?

A Boolean Relaxation

$$\min_{\boldsymbol{z} \in \mathrm{Conv}(\mathcal{Z})} \max_{\boldsymbol{\alpha}} \ \boldsymbol{c}^\top \boldsymbol{z} + h(\boldsymbol{\alpha}) - \sum_i z_i \Omega^\star(\alpha_i)$$

- Solve by sub-gradient descent method, or transform to SOCP.
- Randomly rounding relaxation $\boldsymbol{z}^\star$ according to $z_i \sim \mathrm{Bernoulli}(z_i^\star)$ gives a Boolean vector $\boldsymbol{z}$. How good is it?
- Let $\boldsymbol{z}$ be a random rounding of $\boldsymbol{z}^\star$. Then,

$$0 \leq f(\boldsymbol{z}) - f(\boldsymbol{z}^\star) \leq \epsilon$$

with probability at least

$$1 - |\mathcal{R}| \exp\left(\frac{-\epsilon^2}{\kappa}\right)$$

  - $|\mathcal{R}|$ is number of strictly fractional entries in $\boldsymbol{z}^\star$.
  - κ is a function of $|\mathcal{R}|$, problem data.

How does the approach perform on real data?

Sparse Empirical Risk Minimization Scalability

- For regression $f(\boldsymbol{z})$ is closed form, scales to $100,000$s of features.
- For classification, $f(\boldsymbol{z})$ is cheap, scales to $10,000$s of features.
- Outer-approximation algorithm is more accurate than ElasticNet, MCP, SCAD, and runtimes are comparable to Lasso.
- Code available: github.com/jeanpauphilet.

Sparse Portfolio Selection Scalability

Solves sparse portfolio selection problems with 1000s of securities[2].

Reference	Solution method	Size (no. securities)
Frangioni and Gentile ('09)	Perspective cut+SDP	400
Bonami and Lejeune ('09)	Nonlinear Branch-and-Bound	200
Gao and Li ('13)	SOCP relaxation Branch-and-Bound	300
Cui et al. ('13)	SOCP relaxation Branch-and-Bound	300
Zheng et. al. ('14)	SDP Branch-and-Bound	400
Frangioni et. al. ('16)	Aprox. Proj. Perspective Cut	400
Bertsimas and C-W ('18)	OA with γ-regularization	$3,200$

[2] See Bertsimas & Cory-Wright ('18), for an in-depth implementation.

Network Design Scalability

- $f(\boldsymbol{z})$ obtained by solving a quadratic program.
- Approach solves problems with 100s of nodes.
- Objective value 5% better than CPLEX for small problems, 40% better for large problems.

Main Messages and Highlights

- Don't feel married to big-M! We provide a **non-linear alternative** which often scales as well or better: substituting xz for x and adding a ridge regularizer.

Main Messages and Highlights

- Don't feel married to big-M! We provide a **non-linear alternative** which often scales as well or better: substituting xz for x and adding a ridge regularizer.
- By using strong duality, we derive a saddle-point reformulation, which is **exactly** solvable via an outer-approximation procedure.

Main Messages and Highlights

- Don't feel married to big-M! We provide a **non-linear alternative** which often scales as well or better: substituting xz for x and adding a ridge regularizer.
- By using strong duality, we derive a saddle-point reformulation, which is **exactly** solvable via an outer-approximation procedure.
- Our approach: outer-approximation+warm-start+random rounding **is scalable**.

Thanks for listening!

Questions?

Preprint available at: ryancorywright.github.io

Supplementary Material

Selected References

- Bertsimas, D., Pauphilet, J., Van Parys, B.: Sparse regression: Scalable algorithms and empirical performance. arXiv:1902.06547 (2019)
- Bertsimas, D., Cory-Wright, R.: A scalable algorithm for sparse and robust portfolios. arXiv:1811.00138 (2018)
- Bertsimas, D., Cory-Wright, R., Pauphilet, J: A Unified Approach to Mixed-Integer Optimization: Nonlinear Reformulations and Scalable Algorithms. arXiv:1907.02109 (2019).
- Bertsimas, D., Van Parys, B.: Sparse high dimensional regression: Exact scalable algorithms and phase transitions (2019). Ann. Statist., Accepted (2019).
- Dong, H., Che, K., Linderoth, J: Regularization vs. Relaxation: A conic optimization perspective of statistical variable selection. Opt. Online (2015).
- Frangioni, A., Gentile, M. Perspective cuts for a class of convex 01 mixed integer programs. Math. Prog. **106**:225–236 (2006).
- Gamarnik, D., Zadik, I.: High-dimensional regression with binary coefficients. Estimating squared error and a phase transition. ArXiV:1701.04455 (2017).
- Pilanci, M., Wainwright, M.J., El Ghaoui, L.: Sparse learning via boolean relaxations. Math. Prog. **151**(1), 63–87 (2015).
- Zheng, X., Sun, X., Li, D.: Improving the Performance of MIQP Solvers for Quadratic Programs with Cardinality and Minimum Threshold Constraints: A Semidefinite Program Approach. INFORMS J. Comput. **26**(4):690–703 (2014).