

Learning to Branch: Generalization Guarantees and Limits of Data-Independent Discretization

Maria-Florina Balcan¹

Travis Dick²

Tuomas Sandholm^{1,3,4,5}

Ellen Vitercik⁶

¹Carnegie Mellon University, United States

²Google, United States

³Strategic Machine, Inc., United States

⁴Strategy Robot, Inc., United States

⁵Optimized Markets, Inc., United States

⁶Stanford University, United States

JACM, 2024

Presented by: Hao Su on June 10, 2026

Meet the Authors



Maria-Florina Balcan is the Cadence Design Systems Professor of Computer Science at Carnegie Mellon University, working on machine learning, theoretical computer science, and algorithmic game theory.



Travis Dick is a research scientist at Google and received his PhD in Computer Science from Carnegie Mellon University under the supervision of Maria-Florina Balcan.



Tuomas Sandholm is the Angel Jordan University Professor of Computer Science at Carnegie Mellon University, studying the intersection of artificial intelligence, economics, and operations research.



Ellen Vitercik is an Assistant Professor at Stanford University whose research spans machine learning theory, algorithm design, and the interface between economics and computation.

Outline

Intro

Branch and Bound

Deriving Generalization Bounds

Experiments

Conclusion

Overview

Main Contribution

The first generalization guarantee for data-driven tuning of variable-selection rules in branch-and-bound for IP.

We are going to check in this presentation

- ▶ What this sentence means.
- ▶ Why is this meaningful.
- ▶ Why is this hard & this paper's approach.
- ▶ Extensions & Experiments.

Integer Programs (IPs)



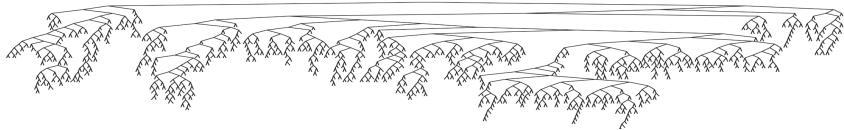
NP-hard

maximize $\mathbf{c} \cdot \mathbf{x}$
subject to $A\mathbf{x} = \mathbf{b}$
 $\mathbf{x} \in \{0,1\}^n$

Branch and Bound (B&B)

- Most widely-used algorithm for IP-solving (CPLEX, Gurobi)
- Recursively partitions search space to find an optimal solution
 - Organizes partition as a tree
- **Many** parameters
 - CPLEX has a 221-page manual describing 135 parameters

“You may need to experiment.”



Intro

Branch and Bound

Algorithm Overview

Variable Selection Policies

Deriving Generalization Bounds

Experiments

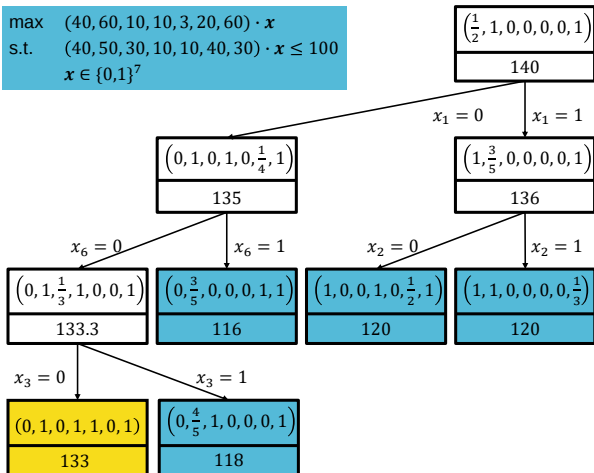
Conclusion

B&B

1. Choose leaf of tree
2. Branch on a variable
3. Fathom leaf if:

- i. LP relaxation solution is integral
- ii. LP relaxation is infeasible
- iii. **LP relaxation solution isn't better than best-known integral solution**

$$\begin{aligned} \max \quad & (40, 60, 10, 10, 3, 20, 60) \cdot x \\ \text{s.t.} \quad & (40, 50, 30, 10, 10, 40, 30) \cdot x \leq 100 \\ & x \in \{0,1\}^7 \end{aligned}$$



B&B

1. Choose leaf of tree
2. Branch on a variable
3. Fathom leaf if:
 - i. LP relaxation solution is integral
 - ii. LP relaxation is infeasible
 - iii. LP relaxation solution isn't better than best-known integral solution

This talk: How to choose which variable?
(Assume every other aspect of B&B is fixed.)

Intro

Branch and Bound

Algorithm Overview

Variable Selection Policies

Deriving Generalization Bounds

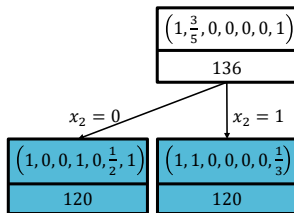
Experiments

Conclusion

Variable selection policies (VSPs)

Score-based VSP:

At leaf Q , branch on variable x_i maximizing
 $\text{score}(Q, i)$



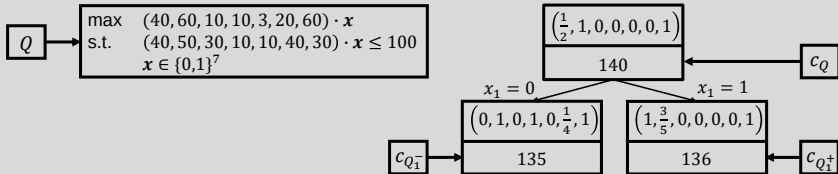
Many options! Little known about which to use when

Variable selection policies

For an IP instance Q :

- Let c_Q be the objective value of its LP relaxation
- Let Q_i^- be Q with x_i set to 0, and let Q_i^+ be Q with x_i set to 1

Example.



Variable selection policies

The linear rule (parameterized by μ) [Linderoth & Savelsbergh, 1999]

Branch on variable x_i maximizing:

$$\text{score}(Q, i) = \mu \min \{c_Q - c_{Q_i^-}, c_Q - c_{Q_i^+}\} + (1 - \mu) \max \{c_Q - c_{Q_i^-}, c_Q - c_{Q_i^+}\}$$

The (simplified) product rule [Achterberg, 2009]

Branch on variable x_i maximizing:

$$\text{score}(Q, i) = (c_Q - c_{Q_i^-}) \cdot (c_Q - c_{Q_i^+})$$

And many more...

Variable selection policies

Given d scoring rules $\text{score}_1, \dots, \text{score}_d$.

Goal: Learn best convex combination $\mu_1 \text{score}_1 + \dots + \mu_d \text{score}_d$.

Our parameterized rule

Branch on variable x_i maximizing:

$$\text{score}(Q, i) = \mu_1 \text{score}_1(Q, i) + \dots + \mu_d \text{score}_d(Q, i)$$

Intro

Branch and Bound

Deriving Generalization Bounds

Discretization Fails

This Paper's Approach

Extensions

Experiments

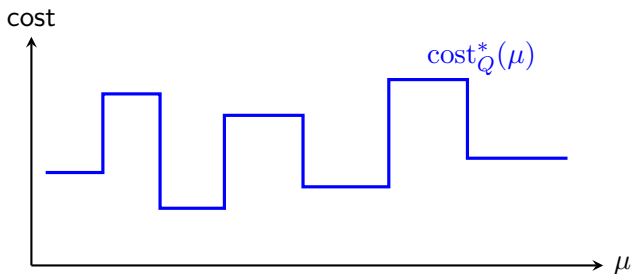
Conclusion

Generalization needs low complexity

Uniformly over all branching parameters μ ,

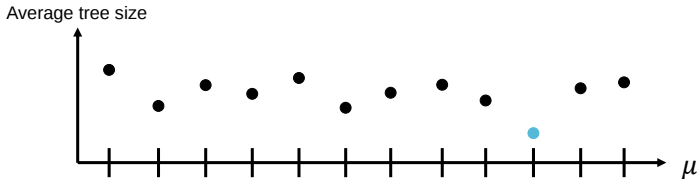
$$\left| \frac{1}{m} \sum_{i=1}^m \text{cost}_{\mu}(Q_i) - \mathbb{E}_{Q \sim \mathcal{D}}[\text{cost}_{\mu}(Q)] \right| \leq \varepsilon(m, \delta).$$

$$\varepsilon \approx \underbrace{\text{complexity}(\mathcal{C})}_{\text{ability to overfit samples}} + \underbrace{\sqrt{\frac{\log(1/\delta)}{m}}}_{\text{concentration}}, \mathcal{C} = \{\text{cost}_{\mu} : \mu \in [0, 1]\}.$$

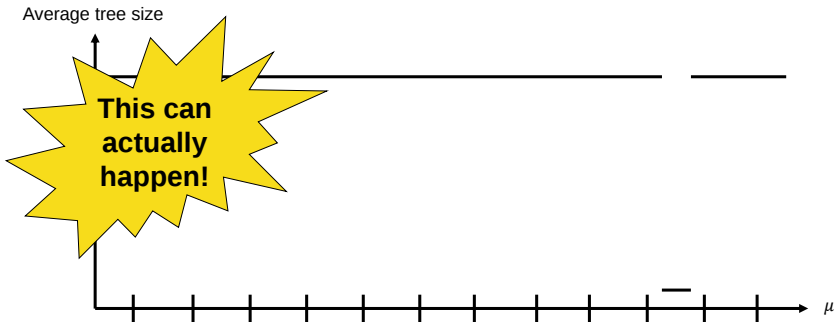


First try: Discretization

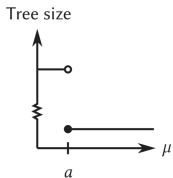
1. Discretize parameter space
2. Receive sample problems from unknown distribution
3. Find params in discretization with best average performance



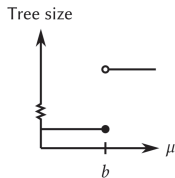
Discretization gone wrong



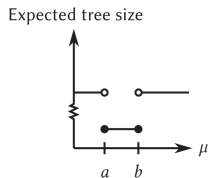
Proof Intuition: Discontinuity



(a) The tree size plot for the instance Q_a as a function of μ .



(b) The tree size plot for the instance Q_b as a function of μ .



(c) The expected tree size plot under the distribution $\mathcal{D}$ as a function of μ .

$$\begin{aligned} \max \quad & \left(1, 2, 3, 4, 5, 0, \frac{3}{2}, 3 - \frac{1}{2\mu^*} \right) \cdot \mathbf{x} \\ \text{s.t.} \quad & \begin{pmatrix} 2 & 2 & 2 & 2 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 2 & 2 \end{pmatrix} \mathbf{x} = \begin{pmatrix} 5 \\ 3 \end{pmatrix} \\ & \mathbf{x} \in \{0, 1\}^8. \end{aligned}$$

$$\begin{aligned} \max \quad & \left(1, 2, 3, 4, 5, 0, \frac{3}{2}, 3 - \frac{1}{2\mu^*} \right) \cdot \mathbf{x} \\ \text{s.t.} \quad & \begin{pmatrix} 2 & 2 & 2 & 2 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 2 & 2 \end{pmatrix} \mathbf{x} = \begin{pmatrix} 5 \\ 3 \end{pmatrix} \\ & \mathbf{x} \in \{0, 1\}^8. \end{aligned}$$

Intro

Branch and Bound

Deriving Generalization Bounds

Discretization Fails

This Paper's Approach

Extensions

Experiments

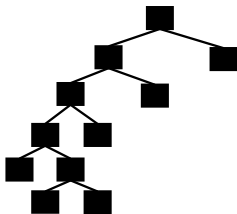
Conclusion

Simple assumption

Exists κ upper bounding the size of largest tree willing to build

Common assumption, e.g.:

- Hutter, Hoos, Leyton-Brown, Stützle, JAIR'09
- Kleinberg, Leyton-Brown, Lucier, IJCAI'17



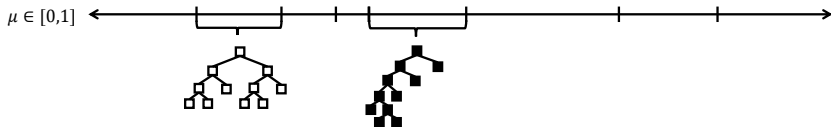
Useful lemma

Much smaller in our experiments!

Lemma: For any two scoring rules and any IP Q ,

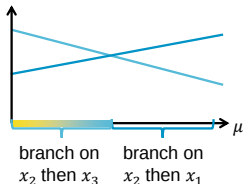
$O((\# \text{ variables})^{\kappa+2})$ intervals partition $[0,1]$ such that:

For any interval $[a, b]$, B&B builds same tree across all $\mu \in [a, b]$



Useful lemma

Lemma: For any two scoring rules and any IP Q ,
 $O((\# \text{ variables})^{\kappa+2})$ intervals partition $[0,1]$ such that:
For any interval $[a,b]$, B&B builds same tree across all $\mu \in [a,b]$



Proof idea.

- Continue dividing $[0,1]$ into intervals s.t.:
In each interval, var. selection order fixed
- Can subdivide only finite number of times
- Proof follows by induction on tree depth

Intro

Branch and Bound

Deriving Generalization Bounds

Discretization Fails

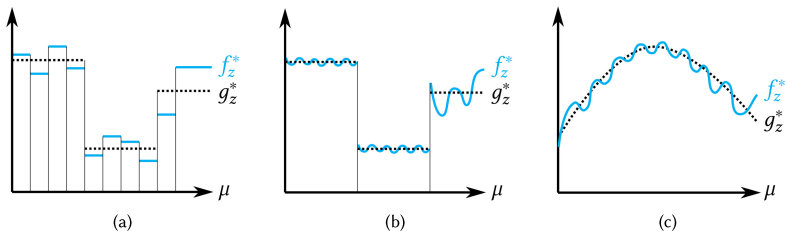
This Paper's Approach

Extensions

Experiments

Conclusion

Data-dependent bounds: approximation vs. complexity



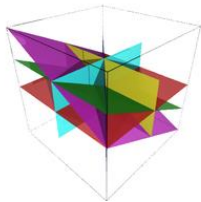
Trade-off

The data-dependent bound has the form

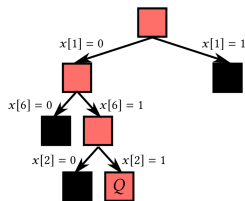
$$\text{generalization error} \lesssim \underbrace{\|F - G\|_{\infty}}_{\text{approximation error}} + \underbrace{\text{complexity}(\mathcal{G}_j)}_{\text{estimation error}}.$$

Useful lemma: higher dimensions

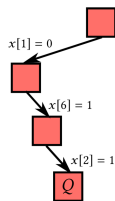
Lemma: For any d scoring rules and any IP,
a set $\mathcal{H}$ of $O((\# \text{ variables})^{\kappa+2})$ hyperplanes partitions $[0,1]^d$ s.t.:
For any connected component R of $[0,1]^d \setminus \mathcal{H}$,
B&B builds the same tree across all $\mu \in R$



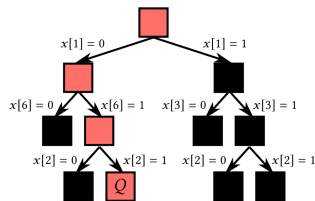
Other Extensions



(a) A B&B search tree $\mathcal{T}$.



(b) The path $\mathcal{T}_Q$ from the root of the tree $\mathcal{T}$ in Figure 2a to the node labeled Q .



(c) Another tree $\mathcal{T}'$ that has the path $\mathcal{T}_Q$ as a rooted subtree.

Path-Wise Scoring Rule for better bounds (by n instead of κ).

Taking log to reduce nonlinear combination of scoring rules of the form $\prod_{i=1}^d \text{score}_i^{\mu_i}$ to convex combinations.

Constraint Satisfaction Problems.

Intro

Branch and Bound

Deriving Generalization Bounds

Experiments

Conclusion

Experiments: What is being demonstrated?

The authors tune the branching score parameter μ and measure the resulting B&B tree size on different problem distributions.

- ▶ **Main empirical observation.** The optimal μ varies substantially across domains, so there is no single universally good branching parameter.
- ▶ **Data-dependent bounds.** They also compare the worst-case generalization bound with the data-dependent bound from Section 3.3. The data-dependent bound is often much smaller, suggesting that the observed dual cost functions have low effective complexity.

Takeaways

- ▶ What IP and B&B are. Why learning is needed.
- ▶ Why discretization failed to give generalization.
- ▶ How the structural traits led to generalization.
- ▶ Data-dependent generalization bounds.
- ▶ What kind of experiments can be carried out.

Recommended to check: Rademacher Complexity.

Questions?