

Develop More Accurate Machine Learning Models with MIP

Dimitris Bertsimas and Daisy Zhuo

Interpretable AI

info@interpretable.ai

September 2020

Outline

- Introduction
- Optimal Trees
- Optimal Feature Selection
- Scalable Implementation with Gurobi Integration
- Success Stories
- Conclusions

- ▶ Interpretable AI is a machine learning software company that builds technologies simultaneously delivering interpretability and state-of-the-art performance.
- ▶ The algorithms are invented and pioneered by the co-founders, and have been successfully applied across a wide variety of industries.



Optimal Imputation

Unlock the full power of data with missing values or quality issues



Optimal Feature Selection

Automatic selection of optimal features from the noise



Optimal Decision Trees

As powerful as black-box artificial intelligence with the interpretability of a single decision tree



Interpretable Matrix Completion

Powerful recommender system that gives detailed explanations for each suggestion

- ▶ Current approaches to machine learning and artificial intelligence like deep learning are black boxes.
- ▶ While many offer accurate predictions, these systems generate outputs based on complex models using billions of parameters with no explanations to why.



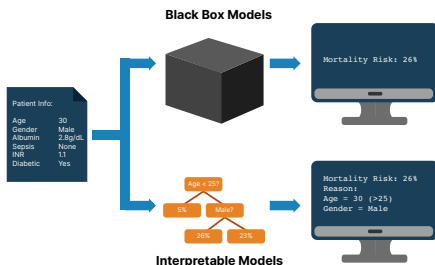
The Apple Card Is Sexist. Blaming the Algorithm Is Proof.

Apple's algorithm seems to be granting women a fraction of their spouses' borrowing limits. It's a troubling example of machine learning's deficiencies.

- ▶ When black boxes go wrong and we cannot inspect it, who should be responsible?
 - ▶ Apple credit card discrimination
 - ▶ Uber self-driving car fatal crashes
- ▶ We need models that leaders and decision makers can stand behind with confidence.

Motivation

- ▶ Traditional interpretable models such as linear regression and decision trees suffer in performance. As a result, practitioners have to choose between **interpretability** and **accuracy**.



- ▶ Can we improve the performance of interpretable models?

- ▶ Many problems in ML/statistics can naturally be expressed as **Mixed integer optimizations (MIO)** problems.
- ▶ MIO in statistics are considered **impractical** and the corresponding problems **intractable**.
- ▶ Heuristics methods are used: Lasso for best subset regression or CART for optimal classification.

- ▶ Speed up between CPLEX 1.2 (1991) and CPLEX 11 (2007): **29,000 times**
- ▶ Gurobi 1.0 (2009) comparable to CPLEX 11
- ▶ Speed up between Gurobi 1.0 and Gurobi 9.0 (2019): **59 times**
- ▶ Total speedup 1991-2019: **1,700,000 times**
- ▶ A MIO that would have taken 16 days to solve 25 years ago can now be solved on the same 25-year-old computer in less than one second.
- ▶ Hardware speed: 93.0 PFlop/s in 2016 vs 59.7 GFlop/s in 1993 **1,600,000 times**
- ▶ Total Speedup: **2.2 Trillion times!**
- ▶ A MIO that would have taken 71,000 years to solve 25 years ago can now be solved in a modern computer in less than one second.

- ▶ Given the dramatically increased power of MIO, **is MIO able to solve** key ML/statistics problems considered intractable a decade ago?

- ▶ How do MIO solutions **compete** with state of the art solutions?

The book provides an original treatment of machine learning (ML) using convex, robust and mixed integer optimization that leads to solutions to central ML problems at large scale that can be found in seconds/minutes, can be certified to be optimal in minutes/hours, and outperform classical heuristic approaches in out-of-sample experiments.

Structure of the book:

- Part I covers robust, sparse, nonlinear, holistic regression and extensions.
- Part II contains optimal classification and regression trees.
- Part III outlines prescriptive ML methods.
- Part IV shows the power of optimization over randomization in design of experiments, exceptional responders, stable regression and the bootstrap.
- Part V describes unsupervised methods in ML: optimal missing data imputation and interpretable clustering.
- Part VI develops matrix ML methods: sparse PCA, sparse inverse covariance estimation, factor analysis, matrix and tensor completion.
- Part VII demonstrates how RL leads to interpretable optimization.

Philosophical principles of the book:

- Interpretability in ML is materially important in real world applications.
- Practical tractability not polynomial solvability leads to real world impact.
- NP-hardness is an opportunity not an obstacle.
- ML is inherently linked to optimization not probability theory.
- Data represents an objective reality; models only exist in our imagination.
- Optimization has a significant edge over randomization.
- The ultimate objective in the real world is prescription, not prediction.

DIMITRIS BERTSIMAS is the Boeing Professor of Operations Research, the co-director of the Operations Research Center and the faculty director of the Master of Business Analytics at the Massachusetts Institute of Technology. He is a member of the National Academy of Engineering, an INFORMS fellow, recipient of numerous research and teaching awards, supervisor of 72 completed and 25 current doctoral theses, and co-founder of ten analytics companies.

JACK DUNN is a co-founding partner of Interpretable AI, a leader of interpretable methods in artificial intelligence. He has a Ph.D. in Operations Research from the Massachusetts Institute of Technology. In his doctoral dissertation he developed optimal classification and regression trees, an important part of this book.



Dynamic Ideas LLC



BERTSIMAS + DUNN

Machine Learning under a Modern Optimization Lens

MACHINE LEARNING UNDER A MODERN OPTIMIZATION LENS

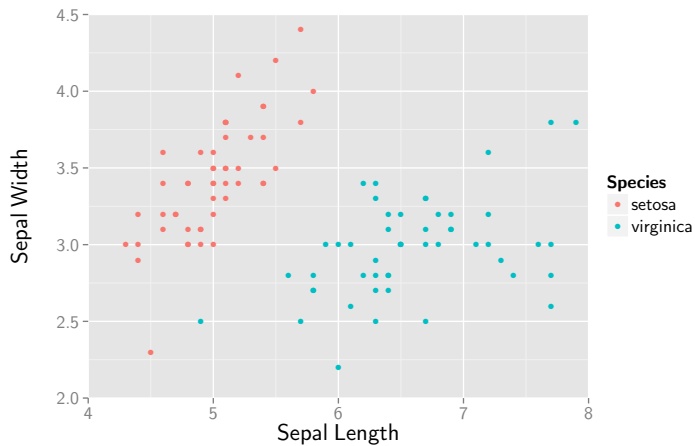
DIMITRIS BERTSIMAS
JACK DUNN

Module 1: Optimal Trees

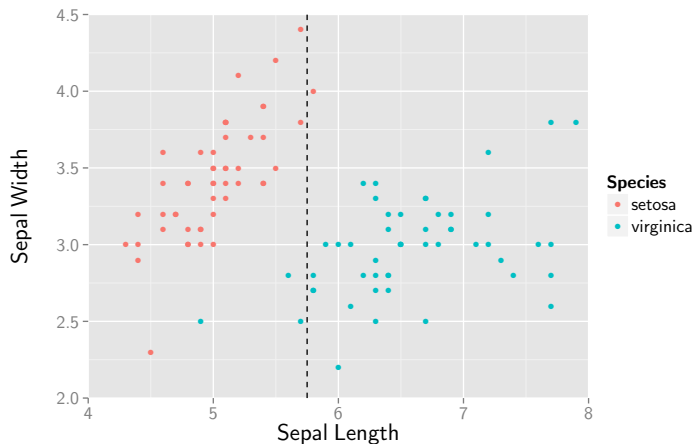
On Interpretability Trees receive an A+

- ▶ Leo Breiman et. al. (1984) introduced CART, a heuristic approach to make predictions (either binary or continuous) from data.
- ▶ Widespread use in academia and industry (~ 47,000 citations!)
- ▶ Let's see an example of Classification Tree, using the Iris flower data set to classify flowers based on four measurements: petal width / height and sepal width / height.

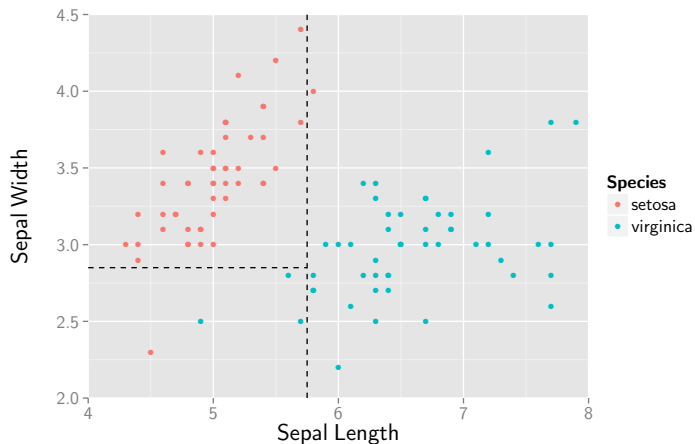
The Iris data set



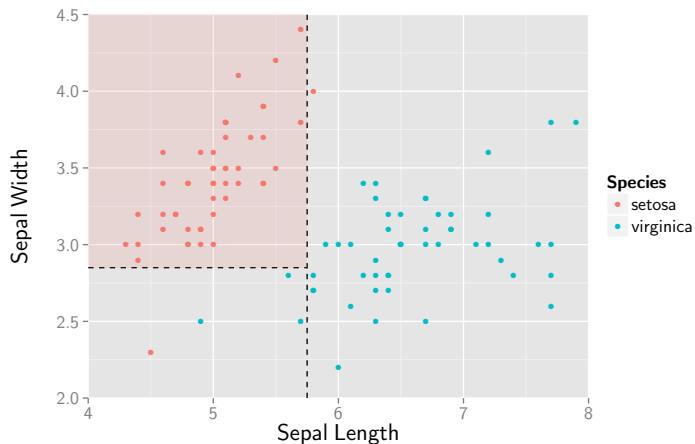
The Iris data set



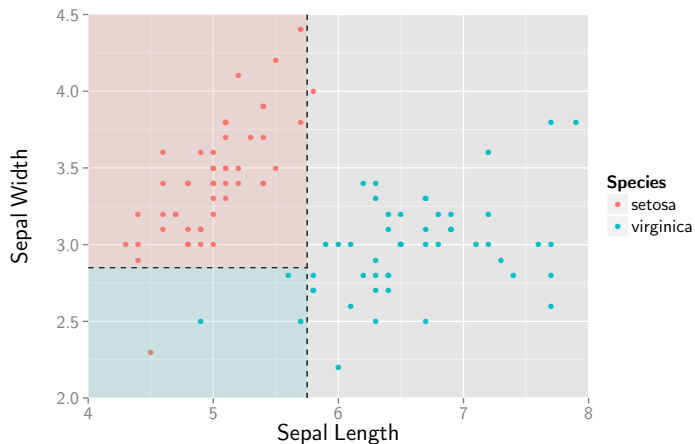
The Iris data set



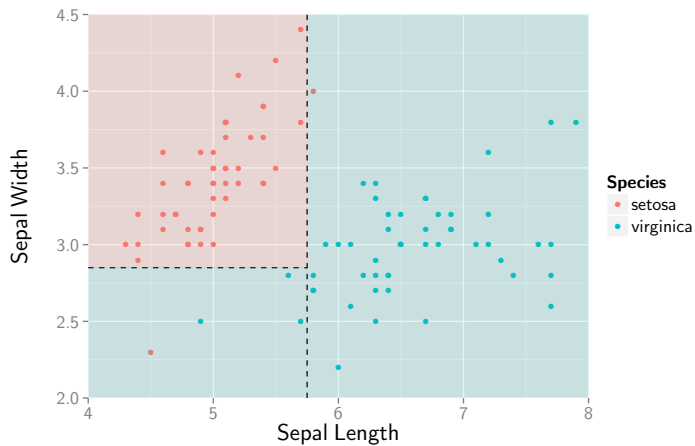
The Iris data set



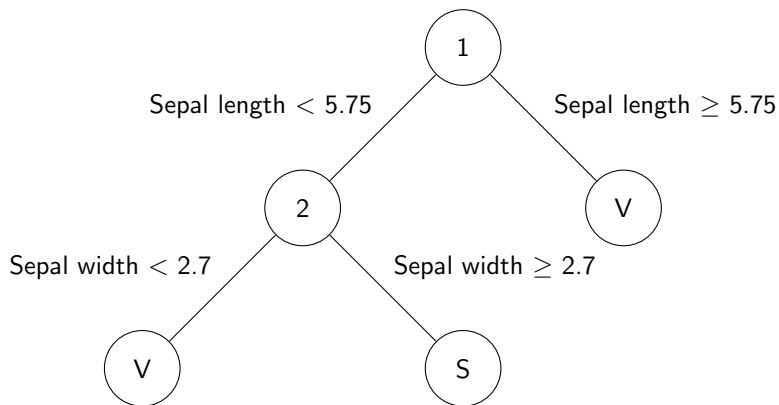
The Iris data set



The Iris data set



The Tree Representation

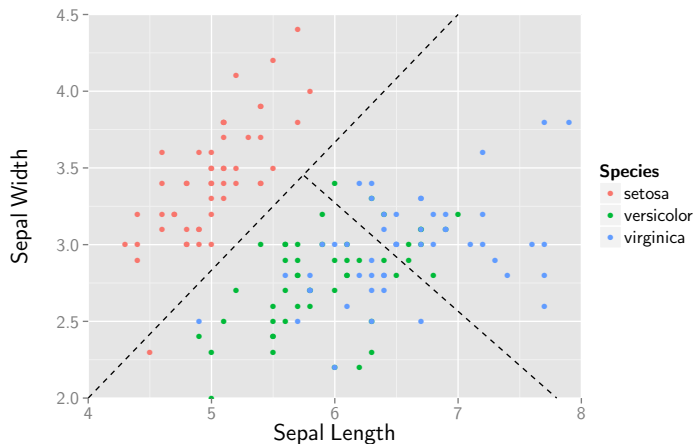


- ▶ CART is fundamentally greedy—it makes a series of locally optimal decisions, but the final tree could be far from optimal

- ▶ *Finally, another problem frequently mentioned (by others, not by us) is that the tree procedure is only one-step optimal and not overall optimal. . . . If one could search all possible partitions . . . the two results might be quite different.*

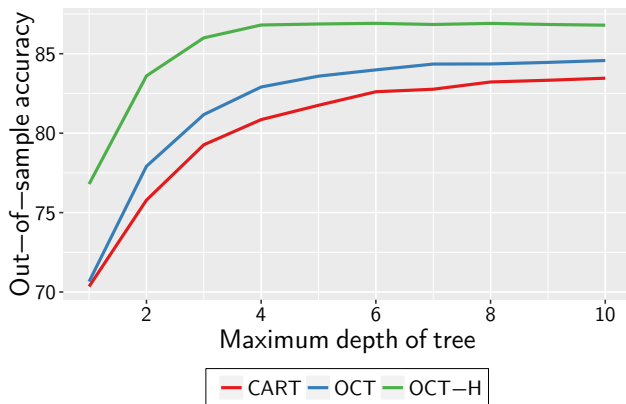
We do not address this problem. At this stage of computer technology, an overall optimal tree growing procedure does not appear feasible for any reasonably sized data set.

- ▶ Use Mixed-Integer Optimization (MIO) and local search to consider the entire decision tree problem at once and solve to obtain the Optimal Tree for both regression and classification.
- ▶ The Algorithms scale with $n = 1,000,000$, $p = 10,000$.
- ▶ **Motivation:** MIO is the natural form for the Optimal Tree problem:
 - ▶ Decisions: Which variable to split on, which label to predict for a region
 - ▶ Outcomes: Which region a point ends up in, whether a point is correctly classified



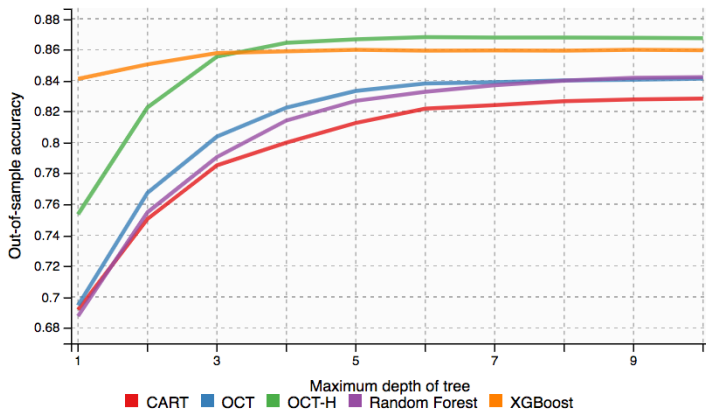
Performance of Optimal Classification Trees

- ▶ Average out-of-sample accuracy across 60 real-world datasets:



Performance of Optimal Classification Trees

- ▶ Average out-of-sample accuracy across 60 real-world datasets:



Variants of Optimal Trees

- ▶ In addition to Optimal Classification Trees, we offer additional flavors of trees tailored to different problem types:



Optimal Classification Trees

Predicts discrete labels - *is this loan likely to default or not?*



Optimal Regression Trees

Predicts continuous/numeric values - *what is the expected revenue for next quarter?*



Optimal Survival Trees

Predicts survival over time - *what is the chance the machine breaks in the next week/month/quarter?*



Optimal Prescriptive Trees

Prescribes personalized optimal decisions - *which marketing outreach strategy is best for each client?*

- ▶ Each one has been used in a number of real-world applications that we will give examples of later.

Module 2: Optimal Feature Selection

- ▶ Linear models such as linear regression and logistic regression are one of the most easily understood and well studied models.
- ▶ However, when the number of features is high, these methods suffer as they don't know how to **select the best subset**.
- ▶ Heuristics such as Lasso and Elastic Net exist, but none solve the exact l_0 problem.

Sparse Regression

- ▶ Problem with regularization

$$\begin{aligned} \min_{\boldsymbol{\beta}} \quad & \frac{1}{2} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \frac{1}{2\gamma} \|\boldsymbol{\beta}\|_2^2 \\ \text{subject to} \quad & \|\boldsymbol{\beta}\|_0 \leq k, \end{aligned}$$

- ▶ Rewrite $\beta_i \rightarrow \beta_i s_i$. Define $\mathbf{S} = \text{diagonal}(\mathbf{s})$.
- ▶ $S_k := \{\mathbf{s} \in \{0, 1\}^P : \mathbf{e}'\mathbf{s} \leq k\}$

$$\min_{\mathbf{s} \in S_k} \left[\min_{\boldsymbol{\beta} \in \mathbb{R}^P} \frac{1}{2} \|\mathbf{y} - \mathbf{X}\mathbf{S}\boldsymbol{\beta}\|_2^2 + \frac{1}{2\gamma} \sum_{i=1}^P s_i \beta_i^2 \right].$$

Sparse Regression

- ▶ Problem with regularization

$$\begin{aligned} \min_{\boldsymbol{\beta}} \quad & \frac{1}{2} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \frac{1}{2\gamma} \|\boldsymbol{\beta}\|_2^2 \\ \text{subject to} \quad & \|\boldsymbol{\beta}\|_0 \leq k, \end{aligned}$$

- ▶ Rewrite $\beta_i \rightarrow \beta_i s_i$. Define $\mathbf{S} = \text{diagonal}(\mathbf{s})$.
- ▶ $S_k := \{\mathbf{s} \in \{0, 1\}^p : \mathbf{e}'\mathbf{s} \leq k\}$

$$\min_{\mathbf{s} \in S_k} \left[\min_{\boldsymbol{\beta} \in \mathbb{R}^p} \frac{1}{2} \|\mathbf{y} - \mathbf{X}\mathbf{S}\boldsymbol{\beta}\|_2^2 + \frac{1}{2\gamma} \sum_{i=1}^p s_i \beta_i^2 \right].$$

- ▶ Solution:

$$\begin{aligned} \min \quad & c(\mathbf{s}) = \frac{1}{2} \mathbf{y}' \left(\mathbb{I}_n + \gamma \sum_j s_j \mathbf{K}_j \right)^{-1} \mathbf{y} \\ \text{subject to} \quad & \mathbf{s} \in S_k. \end{aligned}$$

- ▶ $\mathbf{K}_j := \mathbf{X}_j \mathbf{X}_j'$
- ▶ Binary convex optimization problem

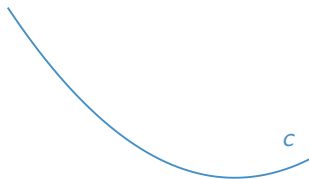
Using Convexity

By convexity of c , for any $\mathbf{s}, \bar{\mathbf{s}} \in S_k$,

$$c(\mathbf{s}) \geq c(\bar{\mathbf{s}}) + \langle \nabla c(\bar{\mathbf{s}}), \mathbf{s} - \bar{\mathbf{s}} \rangle$$

Therefore,

$$c(\mathbf{s}) = \max_{\bar{\mathbf{s}} \in S_k} c(\bar{\mathbf{s}}) + \langle \nabla c(\bar{\mathbf{s}}), \mathbf{s} - \bar{\mathbf{s}} \rangle$$



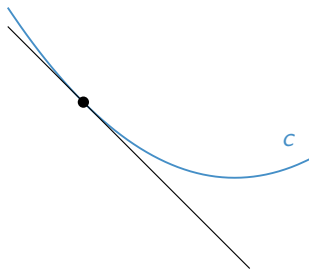
Using Convexity

By convexity of c , for any $\mathbf{s}, \bar{\mathbf{s}} \in S_k$,

$$c(\mathbf{s}) \geq c(\bar{\mathbf{s}}) + \langle \nabla c(\bar{\mathbf{s}}), \mathbf{s} - \bar{\mathbf{s}} \rangle$$

Therefore,

$$c(\mathbf{s}) = \max_{\bar{\mathbf{s}} \in S_k} c(\bar{\mathbf{s}}) + \langle \nabla c(\bar{\mathbf{s}}), \mathbf{s} - \bar{\mathbf{s}} \rangle$$



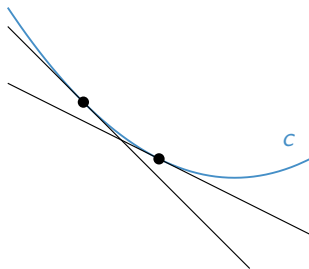
Using Convexity

By convexity of c , for any $\mathbf{s}, \bar{\mathbf{s}} \in S_k$,

$$c(\mathbf{s}) \geq c(\bar{\mathbf{s}}) + \langle \nabla c(\bar{\mathbf{s}}), \mathbf{s} - \bar{\mathbf{s}} \rangle$$

Therefore,

$$c(\mathbf{s}) = \max_{\bar{\mathbf{s}} \in S_k} c(\bar{\mathbf{s}}) + \langle \nabla c(\bar{\mathbf{s}}), \mathbf{s} - \bar{\mathbf{s}} \rangle$$



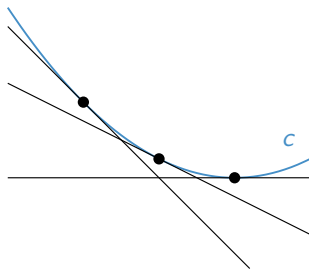
Using Convexity

By convexity of c , for any $\mathbf{s}, \bar{\mathbf{s}} \in S_k$,

$$c(\mathbf{s}) \geq c(\bar{\mathbf{s}}) + \langle \nabla c(\bar{\mathbf{s}}), \mathbf{s} - \bar{\mathbf{s}} \rangle$$

Therefore,

$$c(\mathbf{s}) = \max_{\bar{\mathbf{s}} \in S_k} c(\bar{\mathbf{s}}) + \langle \nabla c(\bar{\mathbf{s}}), \mathbf{s} - \bar{\mathbf{s}} \rangle$$



A Cutting Plane Algorithm

$$c(\mathbf{s}) = \max_{\bar{\mathbf{s}} \in S_k} c(\bar{\mathbf{s}}) + \langle \nabla c(\bar{\mathbf{s}}), \mathbf{s} - \bar{\mathbf{s}} \rangle$$

This leads to a cutting plane algorithm:

1. Pick some $\mathbf{s}_1 \in S_k$ and set $C_1 = \{\mathbf{s}_1\}$.

A Cutting Plane Algorithm

$$c(\mathbf{s}) = \max_{\bar{\mathbf{s}} \in S_k} c(\bar{\mathbf{s}}) + \langle \nabla c(\bar{\mathbf{s}}), \mathbf{s} - \bar{\mathbf{s}} \rangle$$

This leads to a cutting plane algorithm:

1. Pick some $\mathbf{s}_1 \in S_k$ and set $C_1 = \{\mathbf{s}_1\}$.
2. For $t \geq 1$, solve

$$\min_{\mathbf{s} \in S_k} \left[\max_{\bar{\mathbf{s}} \in C_t} c(\bar{\mathbf{s}}) + \langle \nabla c(\bar{\mathbf{s}}), \mathbf{s} - \bar{\mathbf{s}} \rangle \right].$$

A Cutting Plane Algorithm

$$c(\mathbf{s}) = \max_{\bar{\mathbf{s}} \in S_k} c(\bar{\mathbf{s}}) + \langle \nabla c(\bar{\mathbf{s}}), \mathbf{s} - \bar{\mathbf{s}} \rangle$$

This leads to a cutting plane algorithm:

1. Pick some $\mathbf{s}_1 \in S_k$ and set $C_1 = \{\mathbf{s}_1\}$.

2. For $t \geq 1$, solve

$$\min_{\mathbf{s} \in S_k} \left[\max_{\bar{\mathbf{s}} \in C_t} c(\bar{\mathbf{s}}) + \langle \nabla c(\bar{\mathbf{s}}), \mathbf{s} - \bar{\mathbf{s}} \rangle \right].$$

3. If solution \mathbf{s}_t^* to Step 2 has $c(\mathbf{s}_t^*) > \max_{\bar{\mathbf{s}} \in C_t} c(\bar{\mathbf{s}}) + \langle \nabla c(\bar{\mathbf{s}}), \mathbf{s}_t^* - \bar{\mathbf{s}} \rangle$, then set $C_{t+1} := C_t \cup \{\mathbf{s}_t^*\}$ and go back to Step 2.

Cutting plane algorithm can be faster than Lasso.

		Exact T [s]			Lasso T [s]		
		$n = 10k$	$n = 20k$	$n = 100k$	$n = 10k$	$n = 20k$	$n = 100k$
$k = 10$	$p = 50k$	21.2	34.4	310.4	69.5	140.1	431.3
	$p = 100k$	33.4	66.0	528.7	146.0	322.7	884.5
	$p = 200k$	61.5	114.9	NA	279.7	566.9	NA
$k = 20$	$p = 50k$	15.6	38.3	311.7	107.1	142.2	467.5
	$p = 100k$	29.2	62.7	525.0	216.7	332.5	988.0
	$p = 200k$	55.3	130.6	NA	353.3	649.8	NA
$k = 30$	$p = 50k$	31.4	52.0	306.4	99.4	220.2	475.5
	$p = 100k$	49.7	101.0	491.2	318.4	420.9	911.1
	$p = 200k$	81.4	185.2	NA	480.3	884.0	NA

Remark on Complexity

- ▶ Traditional complexity theory suggests that the difficulty of a problem increases with dimension.
- ▶ Sparse regression problem has the property that for small number of samples n , the dual approach takes a large amount of time to solve the problem, but most importantly **the optimal solution does not recover the true signal**.
- ▶ However, for a large number of samples n , dual approach solves the problem extremely fast and recovers 100% of the support of the true regressor β_{true} .

- ▶ The sparsity formulation and cutting plane approach extend to other areas of machine learning. E.g.,
 - ▶ Sparse Classification
 - ▶ Matrix completion with and without side information
 - ▶ Tensor Completion
 - ▶ Sparse Inverse Covariance estimation
 - ▶ Factor Analysis
 - ▶ Sparse PCA

- ▶ Many of which are also part of Interpretable AI's offerings.

Scalable Implementation with Gurobi Integration

Scalable Implementation with Gurobi Integration

- ▶ Code written in Julia, call Gurobi with `JuMP.jl` and `Gurobi.jl`
- ▶ Use the new feature of Gurobi 9.0: Gurobi Compute Server with callbacks
- ▶ Mostly Mixed-Integer-Optimization routine with callbacks
- ▶ Accessible from Python and R as well

Scalable Implementation with Gurobi Integration

- ▶ **Optimal Feature Selection:** Uses Gurobi MIP solver in the exact approach
- ▶ **Interpretable Matrix Completion:** Uses Gurobi MIP solver in the exact approach, currently under development
- ▶ **From Prediction to Prescriptions:** Uses Gurobi solver in the optimization step

Success Stories

- ▶ Built on the foundation of our algorithms, Interpretable AI delivers interpretable and performant end-to-end solutions.



Finance

Personalized
Banking Products
Recommendation



Finance

Marketing
Recommendations
that Maximize Fund
Flow



Finance

Predicting Risk of
Loan Default



Cybersecurity

Improving Malware
Detection in
Cybersecurity



Manufacturing

Understanding
Machine Failures in
Car Manufacturing
Plants



Manufacturing

Interpretable
Predictive
Maintenance for
Turbofans



Manufacturing

Predictive Quality
Solution for Die
Casting in Car
Manufacturing



Manufacturing

Monitoring Hard
Drives in a Data
Center



Manufacturing

Optimal Experiment
Design for
Automotive Testing



Healthcare

Surgical Risk
Calculator: POTTER



Healthcare

Screening
Procedure for
Pediatric Head
Trauma



Healthcare

Mortality Risk
Prediction Tool in
Cancer Patients



Insurance

Optimizing Data
Acquisition



Insurance

Robust Data
Pipeline Design



Retail

Assortment
Optimization with
Consumer
Preference Learning



Real Estate

Pricing for Real
Estate Auctions

- ▶ More details on www.interpretable.ai/solutions

Health Care: Surgical Risk Calculator POTTER

- ▶ Using Optimal Trees, we built a highly accurate and understandable risk predictor validated trusted by top surgeons.
- ▶ Currently used daily in leading hospitals.

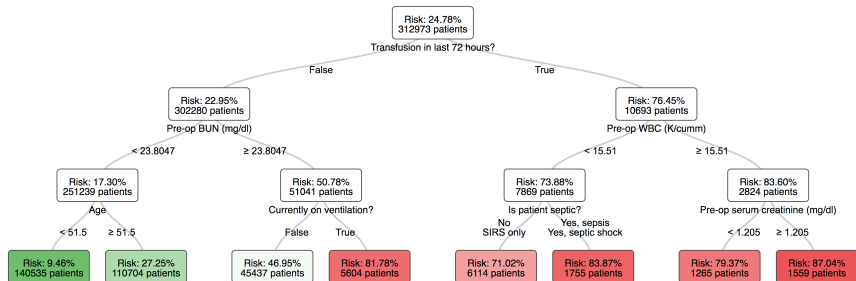


Figure: Decision tree for predicting any complication post surgery.

Health Care: Surgical Risk Calculator POTTER

POTTER Calculator

I would like to predict my patient's 30 day risk of:

- Mortality
- Any complication
- A specific complication

Acute Renal Failure

What is the patient's pre-operative serum creatinine (mg/dl)?

2.5

Is the patient on dialysis or currently requiring dialysis?

NO YES

Is the patient currently on mechanical ventilation?

NO YES

Final risk estimation:
29.36% 576/1962 patients

POTTER Calculator

I would like to predict my patient's 30 day risk of:

- Mortality
- Any complication
- A specific complication

Unplanned Intubation

Does the patient have history of COPD?

NO YES

What is the patient's pre-operative serum albumin (g/dl)?

3

Is the patient septic?

SIRS only

What is the patient's pre-operative PT (seconds)?

16

Final risk estimation:
15.66% 291/1858 patients

Figure: Surgical outcome prediction questionnaire based on Optimal Trees.

Finance: Predicting Risk of Loan Default

- ▶ Credit allocation requires highly understandable models.
- ▶ "Explainable AI" provides post-hoc explanations on existing black box models, but fails to deliver an exact global view.

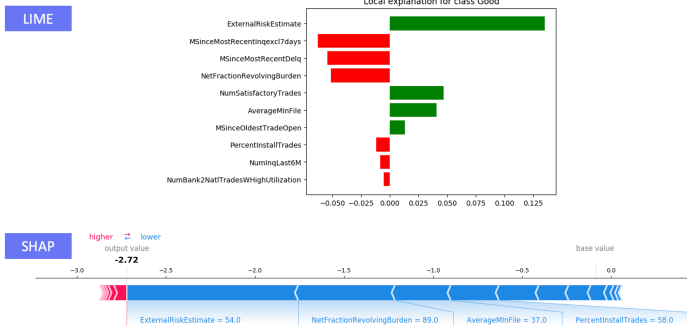
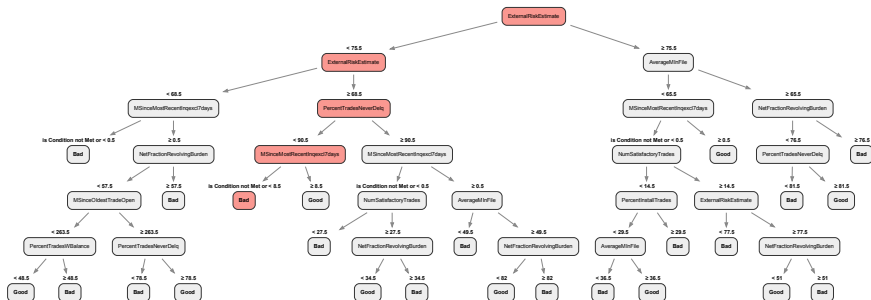


Figure: LIME and SHAP explanation

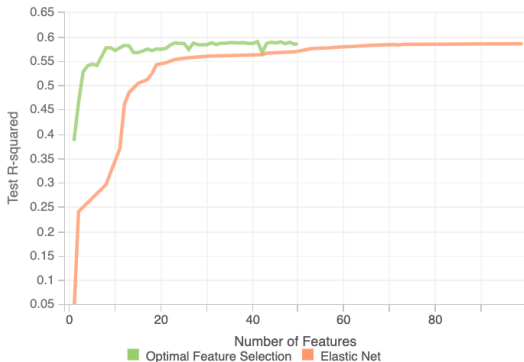
Finance: Predicting Risk of Loan Default

- ▶ Using Optimal Trees, we built a self-explanatory model with comparable performance to XGBoost.
- ▶ We can follow each path and describe exactly why a decision is made.



Manufacturing: Optimal Test Design

- ▶ In automotive testing, we need to know which features are important to design the experiments efficiently.
- ▶ With Optimal Feature Selection, we were able to find the model with peak performance with only 8 features, compared to over 80 with Elastic Net.

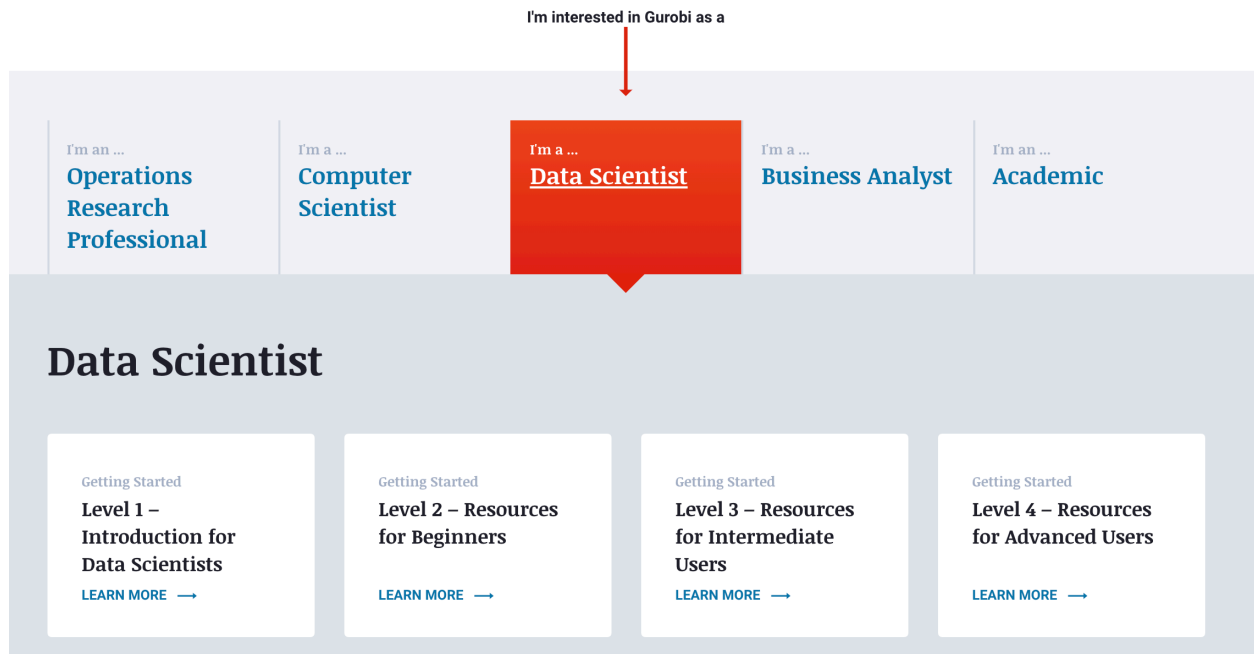


- ▶ Interpretability matters in real world machine learning.
- ▶ MIO solutions have a significant edge over heuristics.
 - ▶ Optimal Trees provide state-of-the-art interpretable solutions.
 - ▶ Optimal Feature Selection solves sparse regression problems with $n = 100,000$ s and $p = 100,000$ to provable optimality in minutes.
- ▶ Interpretable AI builds industrial scale software implementation of the algorithms and delivers exciting applications in medicine, computer security, financial services, manufacturing, among many other fields.

New to Mathematical Optimization?

- **Getting Started with Mathematical Optimization**
 - Visit www.gurobi.com and select 'I am a Data Scientist.'
 - Browse through our "Resource" pages for Beginners, Intermediate users and Advanced users

I'm interested in Gurobi as a



I'm an ...
Operations Research Professional

I'm a ...
Computer Scientist

I'm a ...
Data Scientist

I'm a ...
Business Analyst

I'm an ...
Academic

Data Scientist

Getting Started
Level 1 – Introduction for Data Scientists
[LEARN MORE →](#)

Getting Started
Level 2 – Resources for Beginners
[LEARN MORE →](#)

Getting Started
Level 3 – Resources for Intermediate Users
[LEARN MORE →](#)

Getting Started
Level 4 – Resources for Advanced Users
[LEARN MORE →](#)

Modeling Examples

- **Jupyter Notebook Modeling Examples**
 - [Cell Tower Coverage](#)
 - [Customer Assignment](#)
 - [Facility Location](#)
 - [Feature Selection for Forecasting \(L0-Regression\)](#)
 - [Offshore Wind Farming](#)
 - [Standard Pooling](#)
 - [Traveling Salesman](#)
 - And many more!
- [gurobi.com/examples](https://www.gurobi.com/examples)

 **GUROBI**
OPTIMIZATION

**Build Your Modeling Skills Using
the Gurobi Python API**

Jupyter Notebook Modeling Examples

[Learn More](#)

Next Steps

- **Try Gurobi now**
 - Request a 30-day commercial evaluation license at www.gurobi.com/eval
 - Get a free academic license at <https://www.gurobi.com/academia/>
- **Ask us how you can leverage optimization for your business**
 - Contact us at info@gurobi.com
- **Get ready for Gurobi Optimizer 9.1!**
 - Join us online at the INFORMS Annual Meeting, November 7-13, 2020 for a preview of 9.1.
 - Attend Gurobi sessions and our workshop on Wednesday, Nov. 11, 12-2 PM ET.
 - Visit our virtual booth to chat with experts and for a fun modeling examples challenge.

Thank You



GUROBI
OPTIMIZATION

The World's Fastest Solver