Statistical-Computational Trade-offs for Recursive Adaptive Partitioning Estimators

Jason M. Klusowski
Operations Research and Financial Engineering (ORFE)

PRINCETON
UNIVERSITY

Statistical-Computational Trade-offs for Recursive Adaptive Partitioning Estimators

Yan Shuo Tan
(NUS)

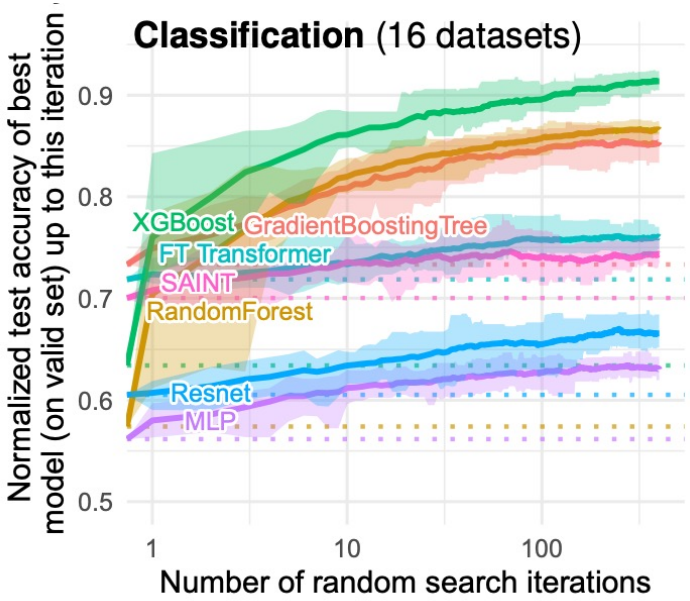
Krishnakumar Balasubramanian
(UC Davis)

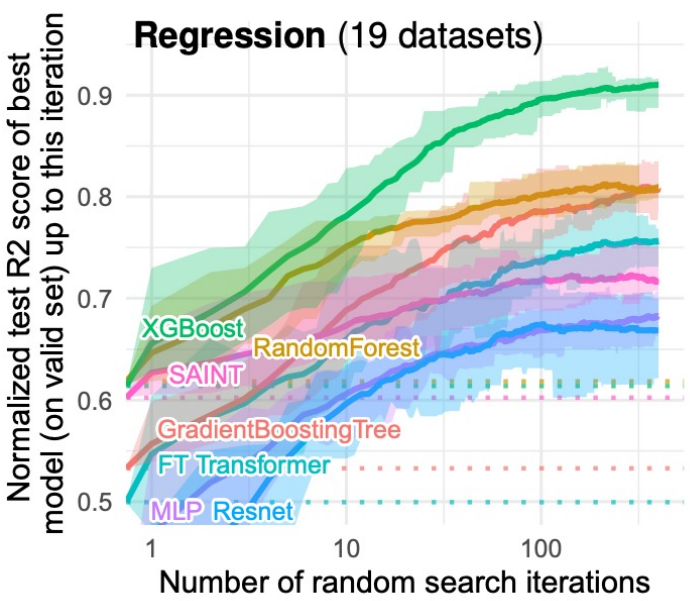
Why do tree-based models still outperform deep learning on tabular data?

Léo Grinsztajn
Soda, Inria Saclay
leo.grinsztajn@inria.fr

Edouard Oyallon
ISIR, CNRS, Sorbonne University

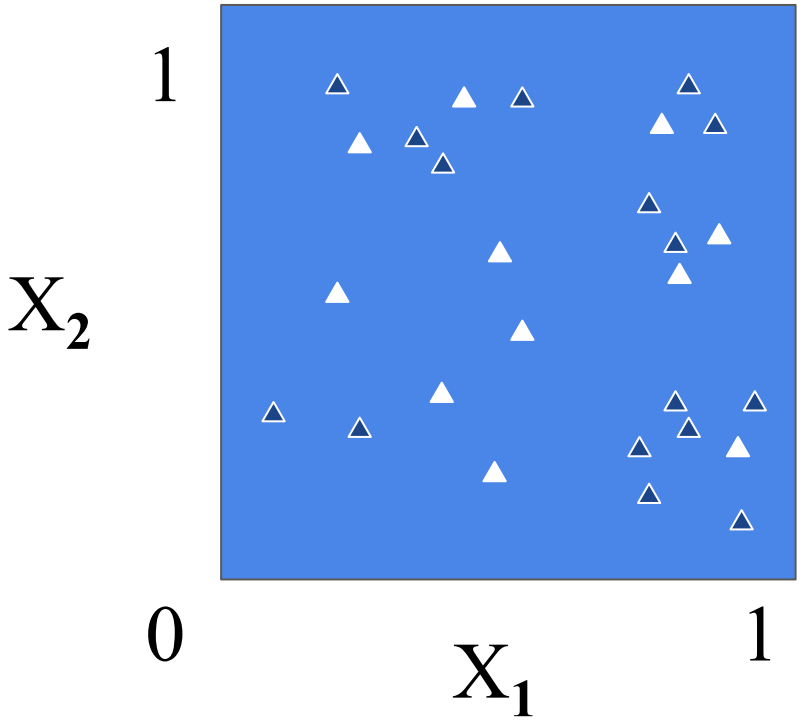
Gaël Varoquaux
Soda, Inria Saclay



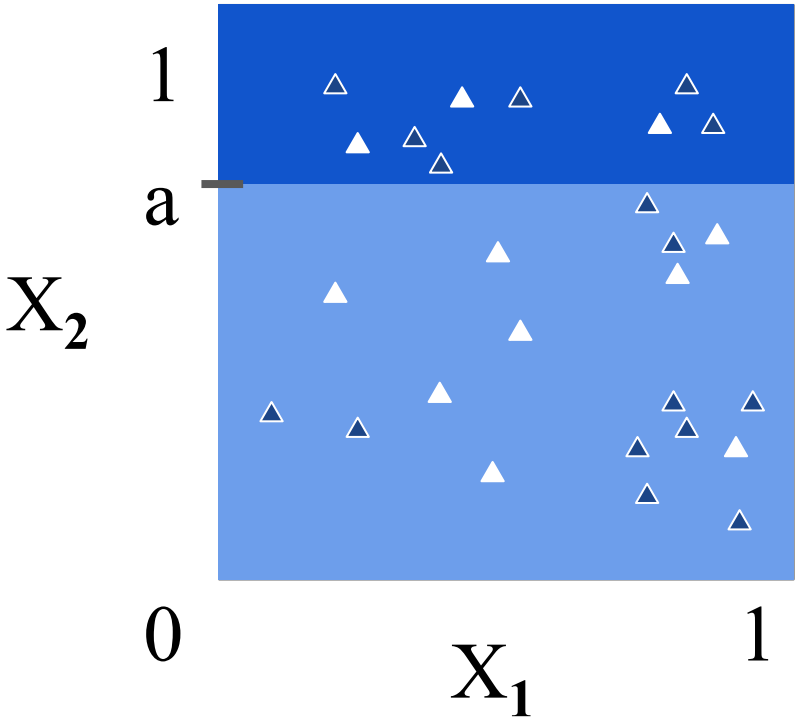


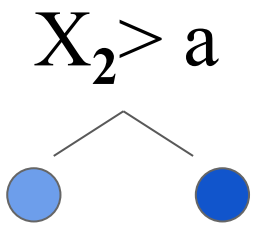
NeurIPS 2022

A regression tree is a **piecewise constant** model obtained from **recursive partitioning** of the covariate space

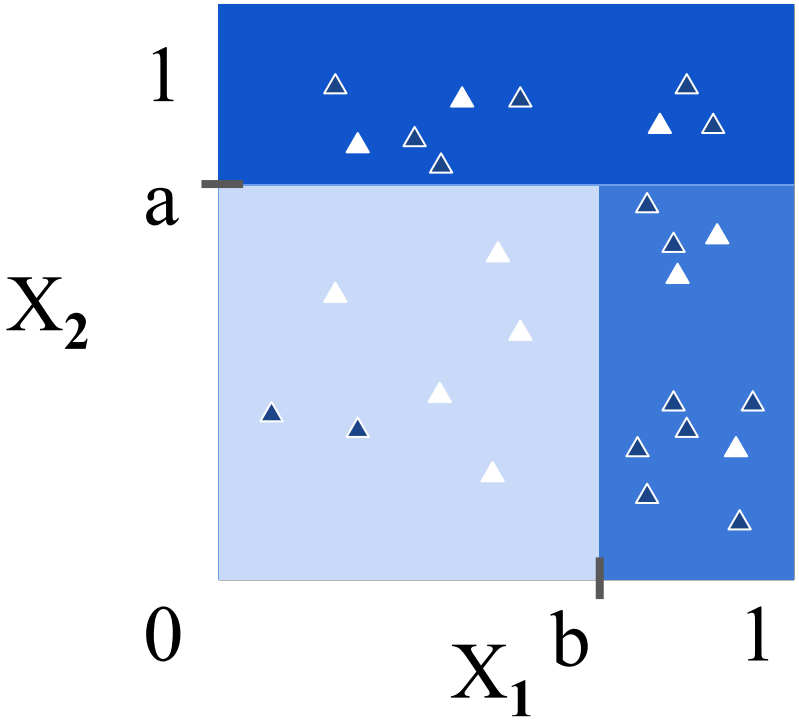


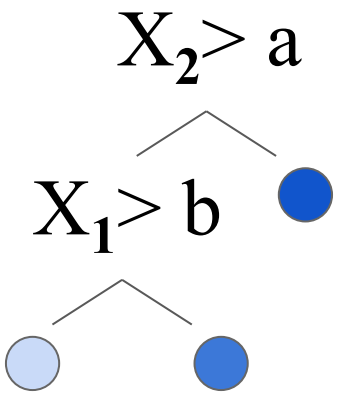
A regression tree is a **piecewise constant** model obtained from **recursive partitioning** of the covariate space





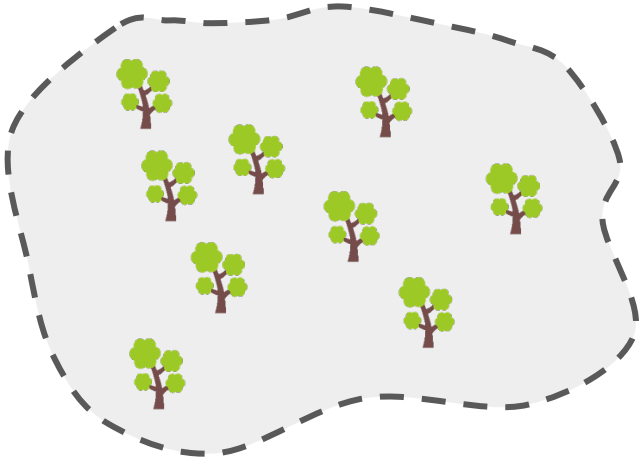
A regression tree is a **piecewise constant** model obtained from **recursive partitioning** of the covariate space





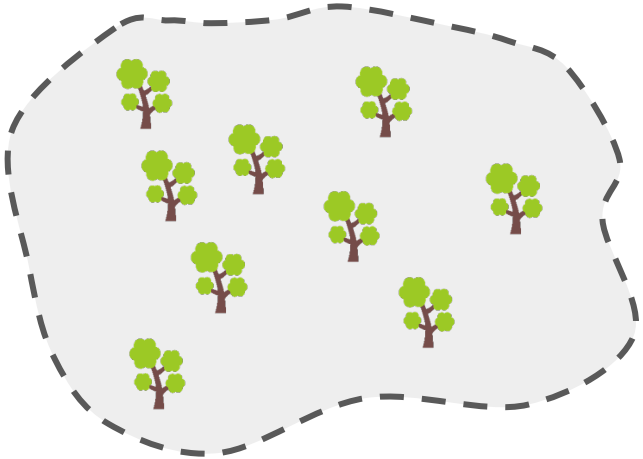
How to grow regression trees?

- Empirical risk minimization (ERM)



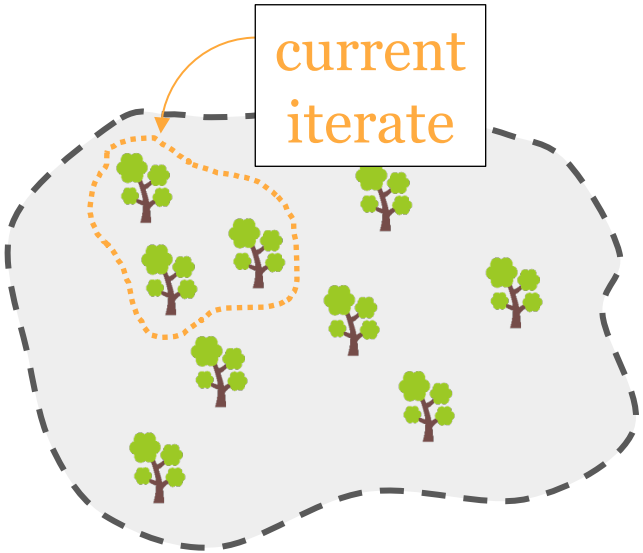
How to grow regression trees?

- Empirical risk minimization (ERM) is NP-hard [Hyafil & Rivest, 1976]



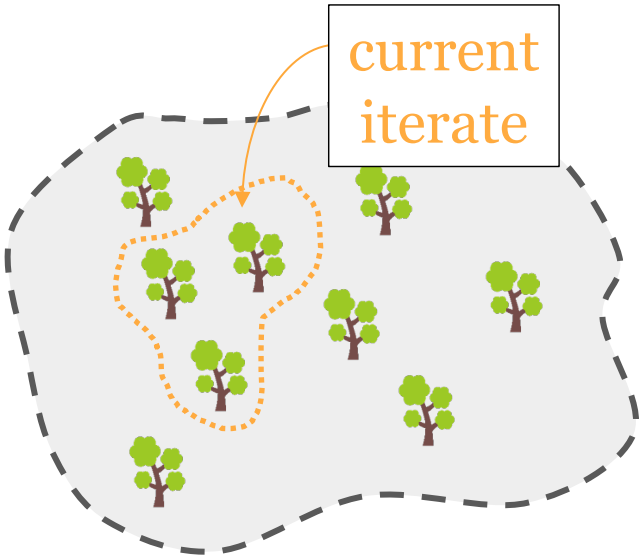
How to grow regression trees?

- Empirical risk minimization (ERM) is NP-hard [Hyafil & Rivest, 1976]
- Greedy algorithms
  - Choose the “best” local update

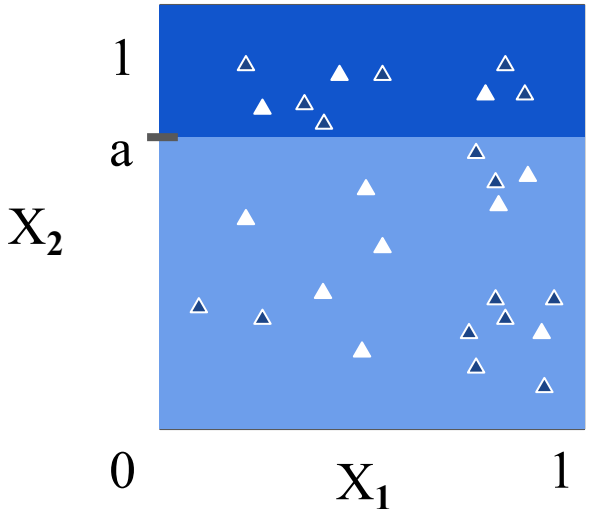


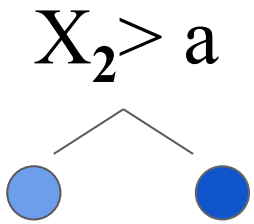
How to grow regression trees?

- Empirical risk minimization (ERM) is NP-hard [Hyafil & Rivest, 1976]
- Greedy algorithms
  - Choose the “best” local update
  - CART [Breiman, Friedman, Olshen, Stone, 1984]
  - Many others: C4.5 [Quinlan, 1993], ID3 [Quinlan, 1986], GUIDE [Loh, 2009],...

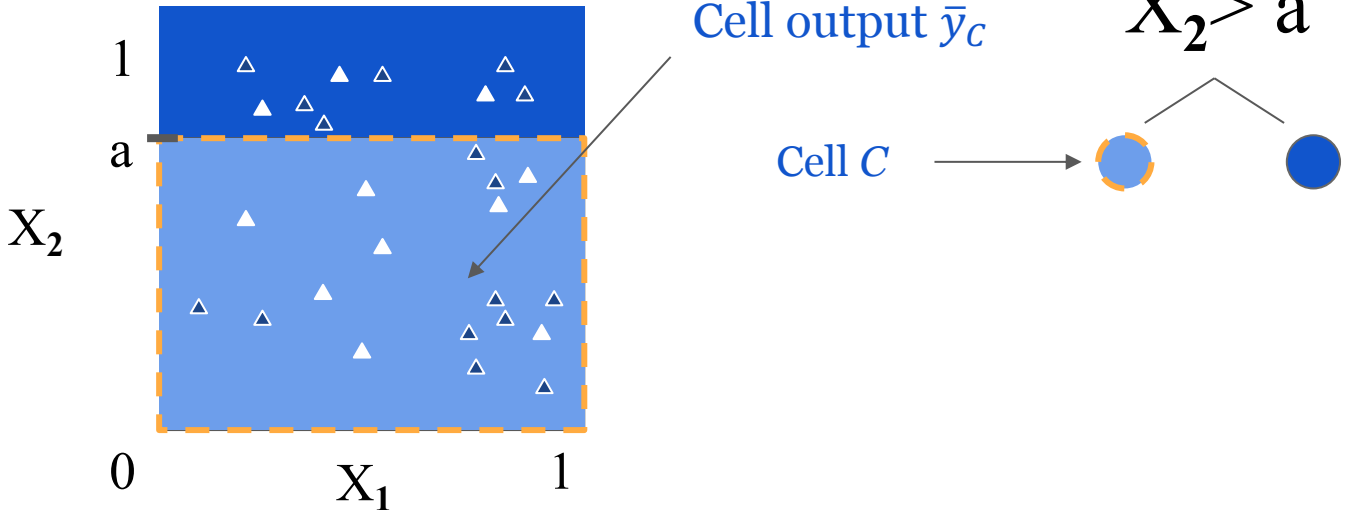


Breiman's CART algorithm

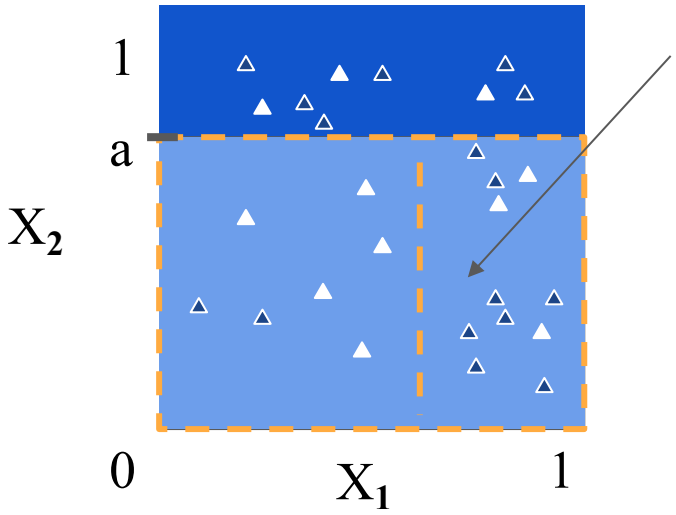




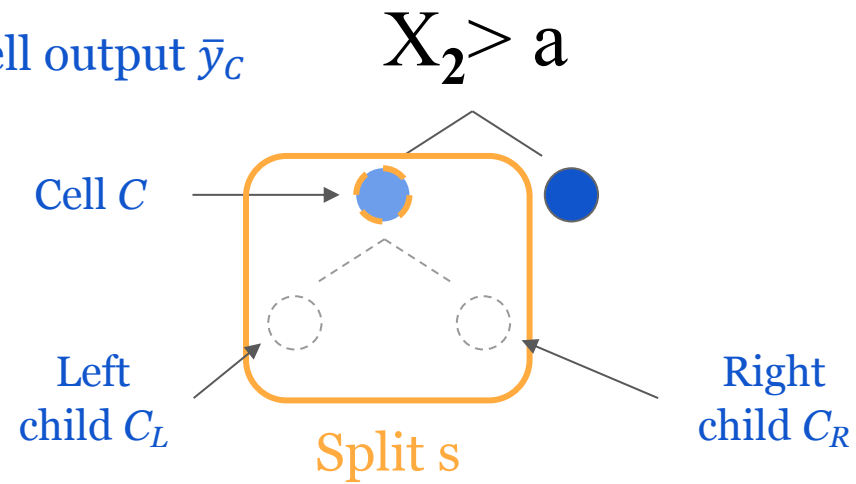
Breiman's CART algorithm



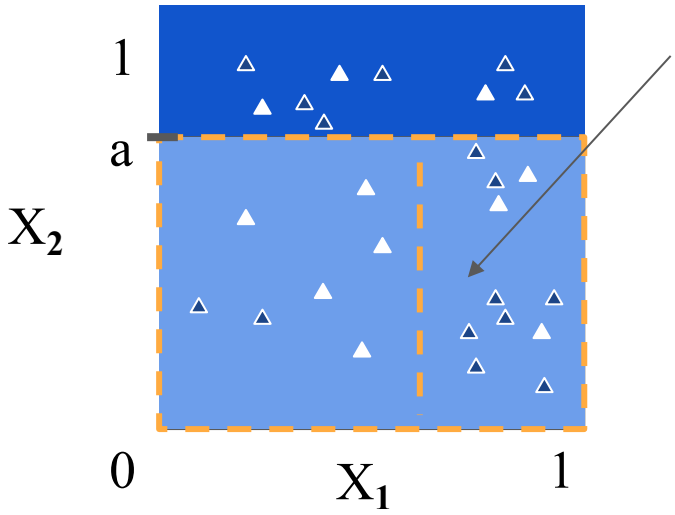
Breiman's CART algorithm



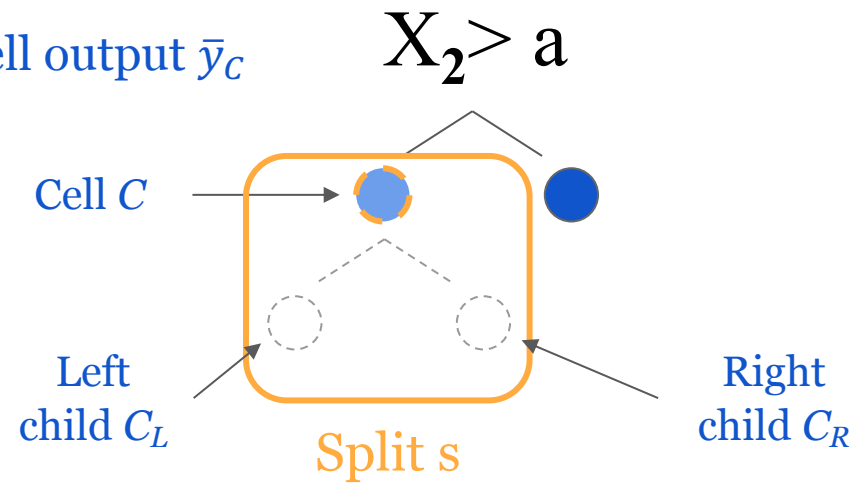
Cell output $\bar{y}_C$



Breiman's CART algorithm



Cell output $\bar{y}_C$

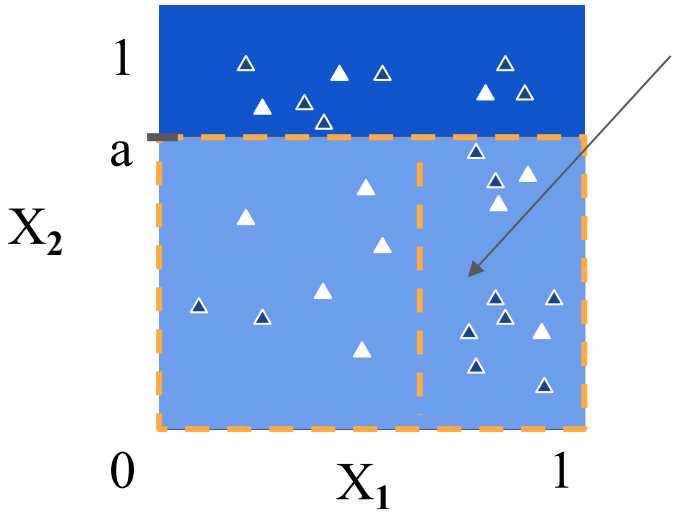


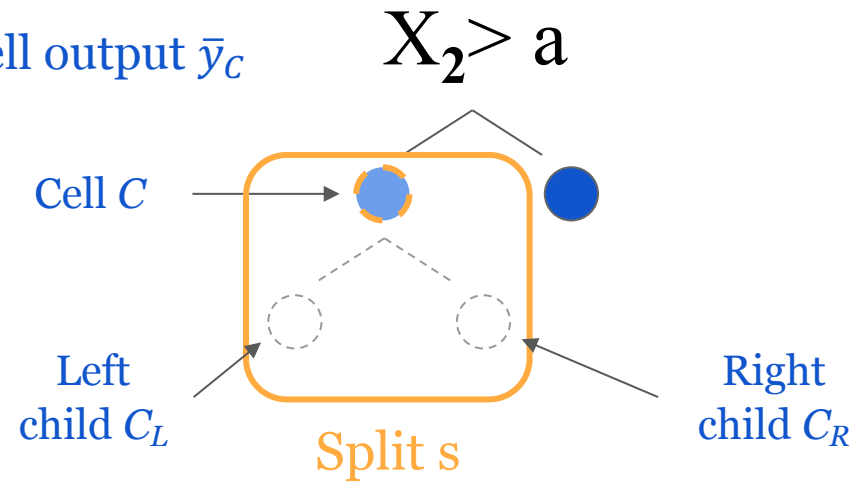
$$\hat{\Delta}(s, \mathcal{C}, \mathcal{D}_n)$$

Impurity decrease

Training dataset

Breiman's CART algorithm

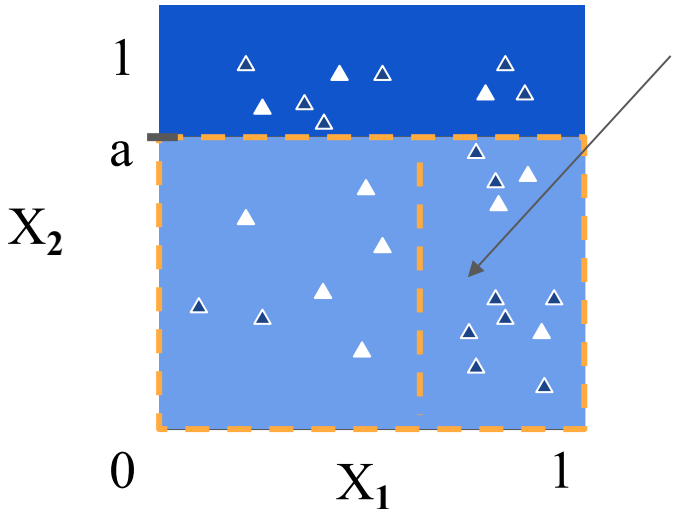




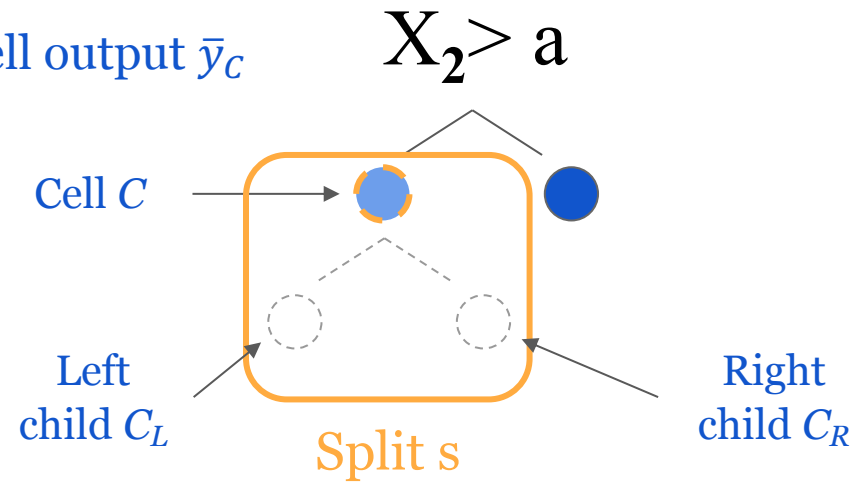
$$\hat{\Delta}(s, \mathcal{C}, \mathcal{D}_n) := \sum_{x_i \in \mathcal{C}} (y_i - \bar{y}_C)^2$$

Impurity decrease

Breiman's CART algorithm



Cell output $\bar{y}_C$



$$\hat{\Delta}(s, \mathcal{C}, \mathcal{D}_n) := \Big(\sum_{x_i \in \mathcal{C}} (y_i - \bar{y}_{\mathcal{C}})^2 - \sum_{x_i \in \mathcal{C}_R} (y_i - \bar{y}_{\mathcal{C}_R})^2 - \sum_{x_i \in \mathcal{C}_L} (y_i - \bar{y}_{\mathcal{C}_L})^2 \Big) / N(\mathcal{C})$$

Impurity decrease

Breiman's CART algorithm

$$\hat{\Delta}(s, \mathcal{C}, \mathcal{D}_n) := \Big(\sum_{x_i \in \mathcal{C}} (y_i - \bar{y}_{\mathcal{C}})^2 - \sum_{x_i \in \mathcal{C}_R} (y_i - \bar{y}_{\mathcal{C}_R})^2 - \sum_{x_i \in \mathcal{C}_L} (y_i - \bar{y}_{\mathcal{C}_L})^2 \Big) / N(\mathcal{C})$$

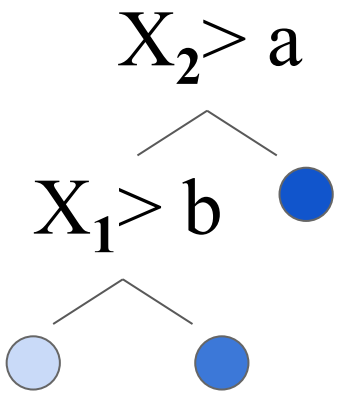
Impurity decrease

$$\approx$$

$$\Delta(s, \mathcal{C}, \mathcal{D}_n) := \mathrm{Var}\{f^*(\mathbf{X}) \mid \mathbf{X} \in \mathcal{C}\} - \frac{\mathbb{P}\{\mathbf{X} \in \mathcal{C}_R\}}{\mathbb{P}\{\mathbf{X} \in \mathcal{C}\}} \mathrm{Var}\{f^*(\mathbf{X}) \mid \mathbf{X} \in \mathcal{C}_R\} - \frac{\mathbb{P}\{\mathbf{X} \in \mathcal{C}_L\}}{\mathbb{P}\{\mathbf{X} \in \mathcal{C}\}} \mathrm{Var}\{f^*(\mathbf{X}) \mid \mathbf{X} \in \mathcal{C}_L\}$$

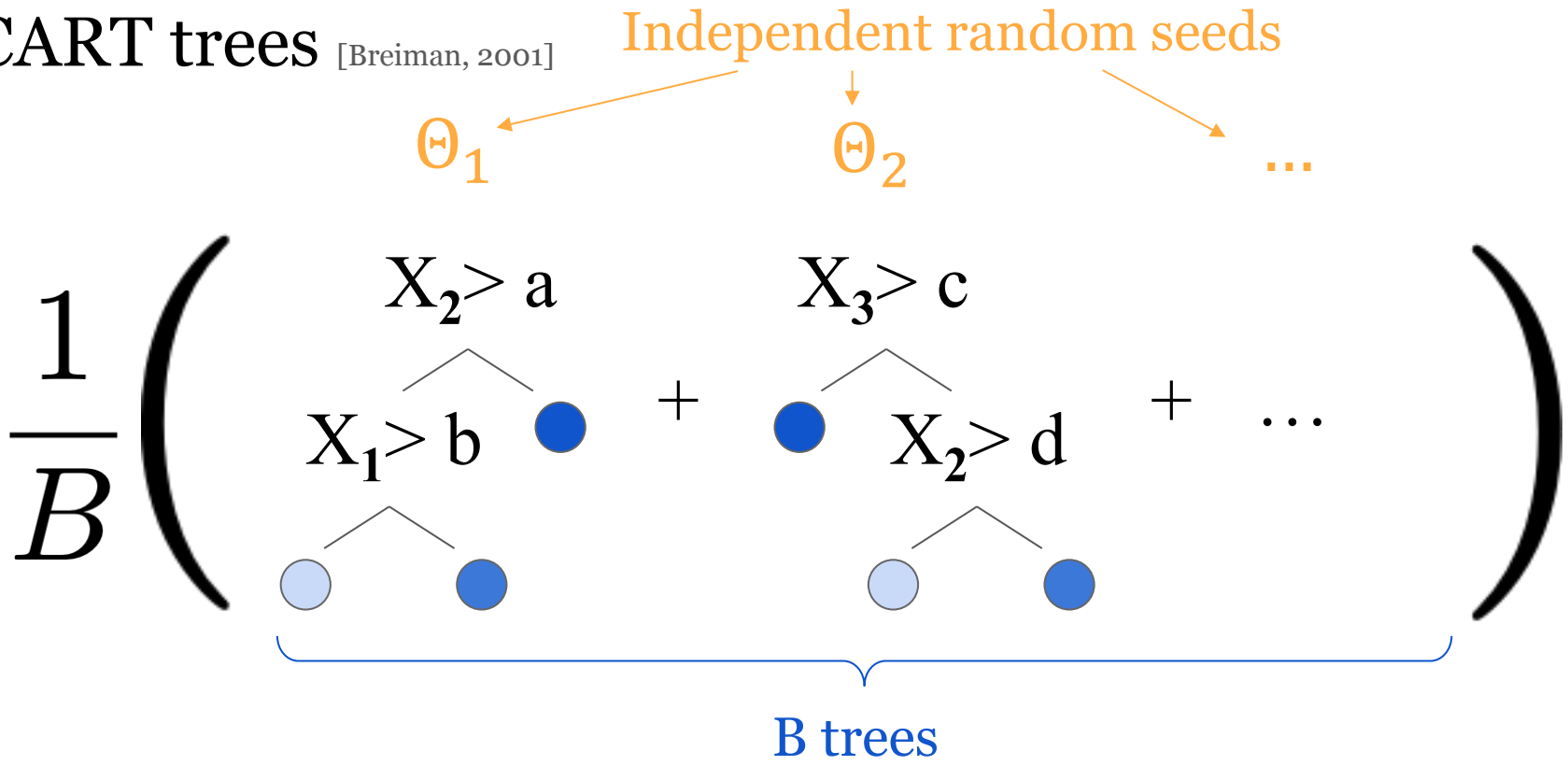
Reduction in residual variance from making the split s

Random forests (RFs) are ensembles of **randomized** CART trees [Breiman, 2001]



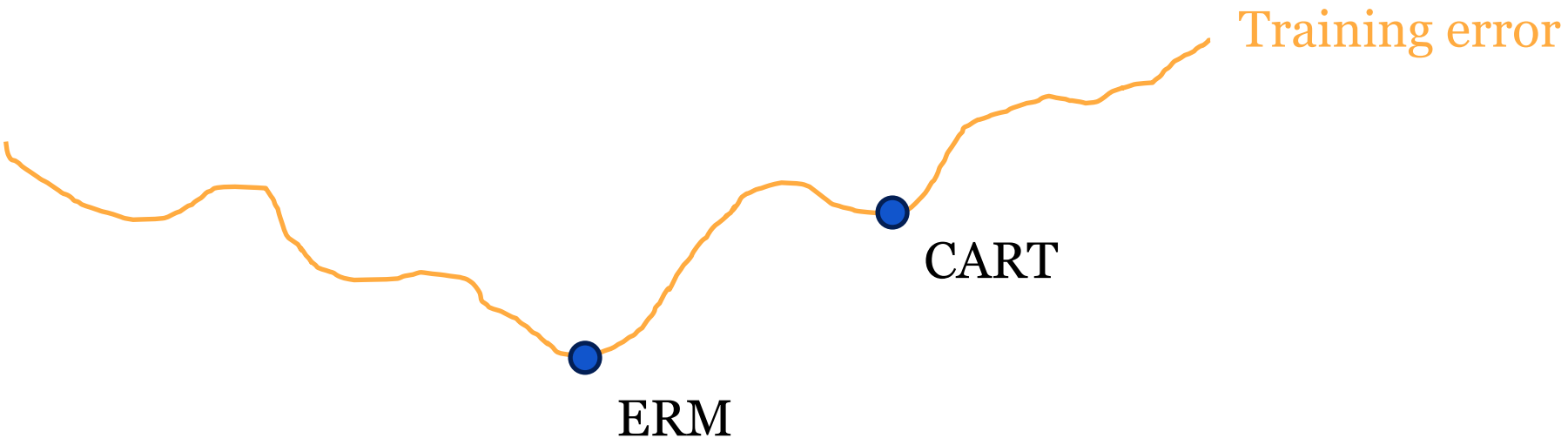
- Each tree is grown on a **bootstrap** resample of $\mathcal{D}_n$
- At each node, the **features are subsampled** before choosing the best split

Random forests (RFs) are ensembles of **randomized** CART trees [Breiman, 2001]



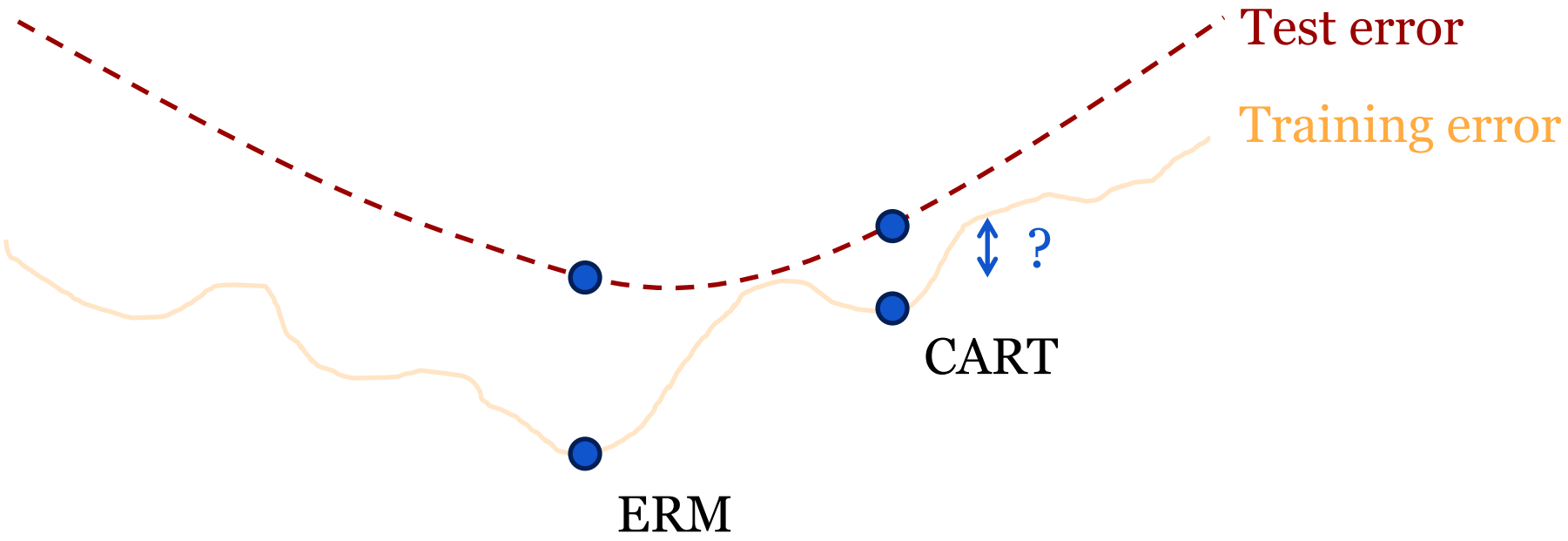
The big question…

The big question…
Is there a **statistical-computational trade-off**?



The big question…

Is there a **statistical-computational trade-off**?



Next step: Identify the right statistical framework

High-dimensional analysis of CART / RFs

- Given i.i.d. data $\mathcal{D}_n = \{(\boldsymbol{X}_i, Y_i)\}_{i=1}^n$
- $Y_i = f^*(\boldsymbol{X}_i) + \epsilon_i, \quad X_i \sim \nu$
- Assume $f^*(\boldsymbol{x}) = f_0^*(\boldsymbol{x}_S)$ for some feature index set S of size s
- Notation: $\mathfrak{R}\left(\hat{f}, f_0^*, d, n\right) = \mathbb{E}_{\mathcal{D}_n,\Theta,\mathbf{X}}\left\{\left(\hat{f}(\mathbf{X}; \mathcal{D}_n, \Theta) - f^*(\mathbf{X})\right)^2\right\}$
- Definition: We say that an estimator $\hat{f}$ is **high-dimensional consistent** if

$$\lim_{d,n\to\infty} \frac{\log n}{d} = 0 \quad \text{and} \quad \lim_{d,n\to\infty} \mathfrak{R}\left(\hat{f}, f_0^*, d, n\right) = 0$$

- Known results: Sparsity + More assumptions $\Rightarrow$ High-dimensional consistency

[Klusowski '20], [Syrganis & Zampetakis '20], [Chi et al. '22], [Mazumder & Wang '24], [Klusowski & Tian '24]

High-dimensional consistency and feature selection

- Definition: We say that an estimator $\hat{f}$ is high-dimensional consistent if

$$\lim_{d,n\to\infty} \frac{\log n}{d} = 0 \quad \text{and} \quad \lim_{d,n\to\infty} \mathfrak{R}\left(\hat{f}, f_0^*, d, n\right) = 0$$

- Average depth of a tree is at most $\log n$

High-dimensional consistency and feature selection

- Definition: We say that an estimator $\hat{f}$ is high-dimensional consistent if

$$\lim_{d,n\to\infty} \frac{\log n}{d} = 0 \quad \text{and} \quad \lim_{d,n\to\infty} \mathfrak{R}\left(\hat{f}, f_0^*, d, n\right) = 0$$

- Average depth of a tree is at most $\log n$ $\implies$ Cannot split on all features
- Hence, **feature selection** is necessary for high-dimensional consistency
- Theoretical results: CART can perform feature selection given **assumptions**

Why do tree-based models still outperform deep learning on tabular data?

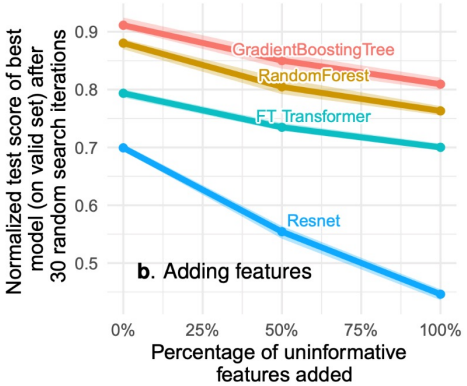
Léo Grinsztajn
Soda, Inria Saclay
leo.grinsztajn@inria.fr

Edouard Oyallon
ISIR, CNRS, Sorbonne University

Gaël Varoquaux
Soda, Inria Saclay

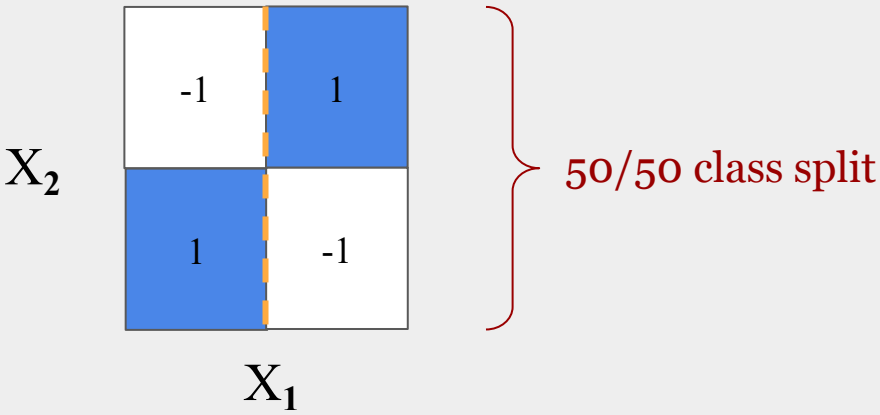
Finding 2: Uninformative features affect more MLP-like neural networks

Tabular datasets contain many uninformative features



Are assumptions necessary?

- Assume binary covariates $\{-1,1\}^d$ with uniform distribution
- Consider the XOR function $f^*(x) = x_1 x_2$ [Syrgkanis & Zampetakis '20], [Mazumder & Wang '24]



Are assumptions necessary?

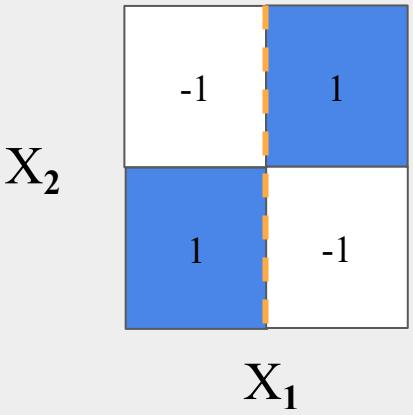
- Assume binary covariates $\{-1,1\}^d$ with uniform distribution
- Consider the XOR function $f^*(x) = x_1 x_2$ [Syrgkanis & Zampetakis '20], [Mazumder & Wang '24]

Are assumptions necessary?

- Assume binary covariates $\{-1,1\}^d$ with uniform distribution
- Consider the XOR function $f^*(x) = x_1 x_2$ [Syrgkanis & Zampetakis '20], [Mazumder & Wang '24]

Are assumptions necessary?

- Assume binary covariates $\{-1,1\}^d$ with uniform distribution
- Consider the XOR function $f^*(x) = x_1 x_2$ [Syrgkanis & Zampetakis '20], [Mazumder & Wang '24]



$$\Delta(1, \{\pm 1\}^d, \mathcal{D}_n) = 0$$

$$\Delta(k, \{\pm 1\}^d, \mathcal{D}_n) = 0, k = 1, 2, \ldots, d$$

$$\Delta(k, \mathcal{C}, \mathcal{D}_n) = 0, k = 1, 2, \ldots, d$$

Unless $\mathcal{C}$ has already split on X_1 or X_2

Marginal signal bottleneck

Are assumptions necessary?

- Assume binary covariates $\{-1,1\}^d$ with uniform distribution
- Consider the XOR function $f^*(x) = x_1 x_2$ [Syrgkanis & Zampetakis '20], [Mazumder & Wang '24]
- CART makes "completely random" splits
- Not high-dimensional consistent
- Until now, no formal proof
- No generalization beyond this example

Address these issues + more

Generalizing the XOR function

- Why is XOR hard?
  - “Pure interaction”, contains no marginal information
  - On the other hand, CART uses only marginal information to determine splits
- Other pure interactions: Boolean monomials
  - For any $S \subset \{1, 2, \ldots, d\}, \chi_S(x) = \prod_{i \in S} x_i$
- Proposition (Fourier basis for Boolean cube):
  - Any function $f : \{\pm 1\}^d \to \mathbb{R}$ has a unique decomposition $f(x) = \sum_{S \subset [d]} \alpha_S \chi_S(x)$
  - ANOVA decomposition with contrast χ_S^2 and effect size α_S
  - Impose heredity constraint on the pattern of interactions

Generalizing the XOR function [Abbe et al., '21]; [Abbe et al., '22]

Definition: We say that a function $f(x) = \sum_{S \subset [d]} \alpha_S \chi_S(x)$ has the *Staircase Property* if $S_1 \subset S_2 \subset \cdots \subset S_k, \quad |S_1| = 1, \quad |S_j \backslash S_{j-1}| = 1$

- Examples: $x_1 + x_1 x_2$ ✅ $x_1 x_2$ ❌ $x_1 + x_1 x_2 x_3$

Definition: We say that a function $f(x) = \sum_{S \subset [d]} \alpha_S \chi_S(x)$ has the *Merged Staircase Property (MSP)* if $|S_j \setminus \cup_{i=1}^{j-1} S_i| \leq 1, \quad j = 1, 2, \ldots, k$

- Examples: $x_1 + x_1 x_2$ ✅ $x_1 x_2$ ❌ $x_1 + x_2 + x_1 x_2 x_3$ $x_1 + x_1 x_2 x_3$

Main results

Definition: We say that a function $f(x) = \sum_{S \subset [d]} \alpha_S \chi_S(x)$ has the *Merged Staircase Property (MSP)* if $\left|S_j \setminus \cup_{i=1}^{j-1} S_i\right| \leq 1, \quad j = 1, 2, \ldots, k$

Theorem (informal): Suppose f_0^* depends only on s covariates. Then

- (Necessity) If f^* does not satisfy MSP, then $\Re\left(\hat{f}_{CART}, f_0^*, d, n\right) = \Omega(1)$ whenever $n = \exp(O(d))$.

Main results

Definition: We say that a function $f(x) = \sum_{S \subset [d]} \alpha_S \chi_S(x)$ has the *Merged Staircase Property (MSP)* if $\left|S_j \setminus \cup_{i=1}^{j-1} S_i\right| \leq 1, \quad j = 1, 2, \ldots, k$

Theorem (informal): Suppose f_0^* depends only on s covariates. Then

- (Necessity) If f^* does not satisfy MSP, then $\mathfrak{R}\left(\hat{f}_{CART}, f_0^*, d, n\right) = \Omega(1)$ whenever $n = \exp(O(d))$.
- (Near sufficiency) If f^* satisfies MSP and Fourier coefficients are generic, then $\mathfrak{R}\left(\hat{f}_{CART}, f_0^*, d, n\right) = O(2^s \log d / n)$.

Main results

Definition: We say that a function $f(x) = \sum_{S \subset [d]} \alpha_S \chi_S(x)$ has the *Merged Staircase Property (MSP)* if $\left|S_j \setminus \cup_{i=1}^{j-1} S_i\right| \leq 1, \quad j = 1, 2, \ldots, k$

Theorem (informal): Suppose f_0^* depends only on s covariates. Then

- (Necessity) If f^* does not satisfy MSP, then $\mathfrak{R}\left(\hat{f}_{CART}, f_0^*, d, n\right) = \Omega(1)$ whenever $n = \exp(O(d))$.
- (Near sufficiency) If f^* satisfies MSP and Fourier coefficients are generic, then $\mathfrak{R}\left(\hat{f}_{CART}, f_0^*, d, n\right) = O(2^s \log d / n)$.

Furthermore, regardless of whether f^* satisfies MSP, $\mathfrak{R}\left(\hat{f}_{ERM}, f_0^*, d, n\right) = O(2^s \log d / n)$.

Main results: Sample complexities

	MSP	Non-MSP
CART	$O(2^s \log d)$	$\exp(\Omega(d))$
ERM	$O(2^s \log d)$	$O(2^s \log d)$
Non-adaptive	$\exp(\Omega(d))$	$\exp(\Omega(d))$

- Establish a **statistical-computational trade-off**

Main results: Sample complexities

	MSP	Non-MSP
CART	$O(2^s \log d)$	$\exp(\Omega(d))$
ERM	$O(2^s \log d)$	$O(2^s \log d)$
Non-adaptive	$\exp(\Omega(d))$	$\exp(\Omega(d))$

- Establish a statistical-computational trade-off
- **Characterize** the regression functions for which CART is high-dimensional consistent

Main results: Sample complexities

	MSP	Non-MSP
CART	$O(2^s \log d)$	$\exp(\Omega(d))$
ERM	$O(2^s \log d)$	$O(2^s \log d)$
Non-adaptive	$\exp(\Omega(d))$	$\exp(\Omega(d))$

- Establish a statistical-computational trade-off
- Characterize the regression functions for which CART is high-dimensional consistent
    - Lower bounds hold more broadly for **RFs** and other greedy trees and ensembles
    - Lower bounds hold when there is **no noise**
    - Lower bounds have **robust** versions that hold for MSP functions

Head-to-head comparison with neural networks

- Recent work in neural network theory: What types of functions are learnable by 2-layer neural networks **optimized using SGD**? [Abbe et al. '21], [Abbe et al., '22], [Barak et al., '22], [Abbe et al., '23], [Suzuki et al., '23], [Glasgow, '24], [Kou et al., '24], [Nichani et al., '23]*, [Fu et al., '24]*, …
- [Abbe et al. '22] studied binary features under the **mean field regime**

  - Fully connected, large width, (small) constant step size, online SGD for O(d) iterations [one sample per iteration]
  - SGD dynamics can be approximated by nonlinear PDE [Mei et al. '18]

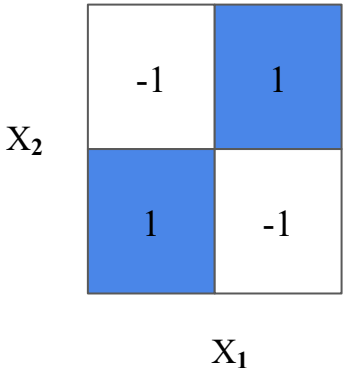
Head-to-head comparison with neural networks

- Recent work in neural network theory: What types of functions are learnable by 2-layer neural networks **optimized using SGD**? [Abbe et al. '21], [Abbe et al., '22], [Barak et al., '22], [Abbe et al., '23], [Suzuki et al., '23], [Glasgow, '24], [Kou et al., '24], [Nichani et al., '23]*, [Fu et al., '24]*, ...
- [Abbe et al. '22] studied binary features under the **mean field regime**
  - MSP characterizes learnability
  - (Necessity) Learnable if f^* satisfies* MSP and Fourier coefficients are generic
  - (Near sufficiency) Not learnable if f^* does not satisfy MSP

Head-to-head comparison with neural networks

	MSP	Non-MSP
CART	$O(2^s \log d)$	$\exp(\Omega(d))$
ERM	$O(2^s \log d)$	$O(2^s \log d)$
Non-adaptive	$\exp(\Omega(d))$	$\exp(\Omega(d))$
Two-layer NNs [Abbe et al. '22]	$O(d)$	$\omega(d)$

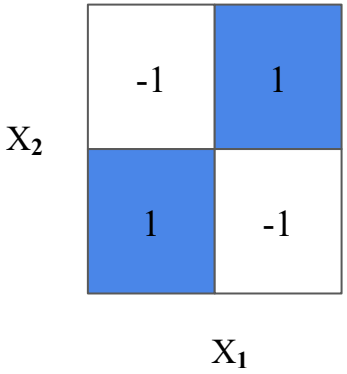
Why are lower bounds hard? E.g., XOR



- Naive strategy: Prove that CART never splits on X_1 or X_2
  - Need concentration for impurity decrease values

W.h.p. $\left|\hat{\Delta}(k, \mathcal{C}, \mathcal{D}_n) - \Delta(k, \mathcal{C}, \mathcal{D}_n)\right| = O\left(n^{-1/2}\right)$ ✅

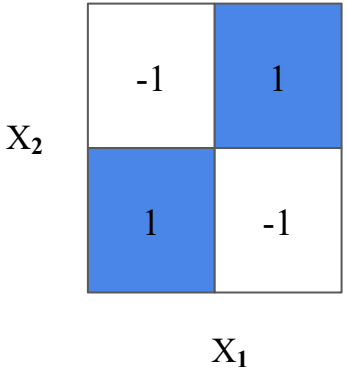
Why are lower bounds hard? E.g., XOR



- Naive strategy: Prove that CART never splits on X_1 or X_2
  - Need concentration for impurity decrease values **for all possible splits**

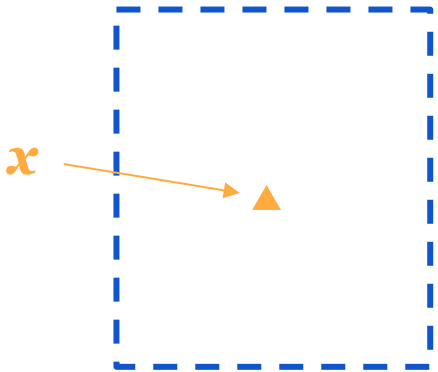
W.h.p. $\sup_{\mathcal{C},k} \left|\hat{\Delta}(k,\mathcal{C},\mathcal{D}_n) - \Delta(k,\mathcal{C},\mathcal{D}_n)\right| = O\left(n^{-1/2}\right)$ ✗

Why are lower bounds hard? E.g., XOR



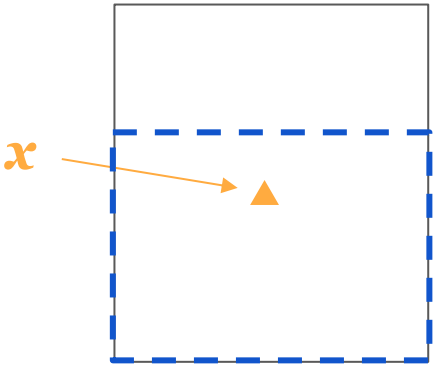
- Naive strategy: Prove that CART never splits on X_1 or X_2
  - Need concentration for impurity decrease values for all possible splits
- New strategy: Prove that **for a single root-to-leaf path**, CART never splits on X_1 or X_2
  - New interpretation of CART root-to-leaf path as a stochastic process
  - Use coupling and symmetry

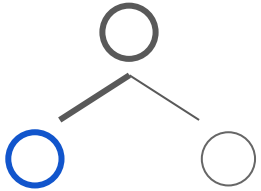
Root-to-leaf path is a stochastic process



C_o

Root-to-leaf path is a stochastic process

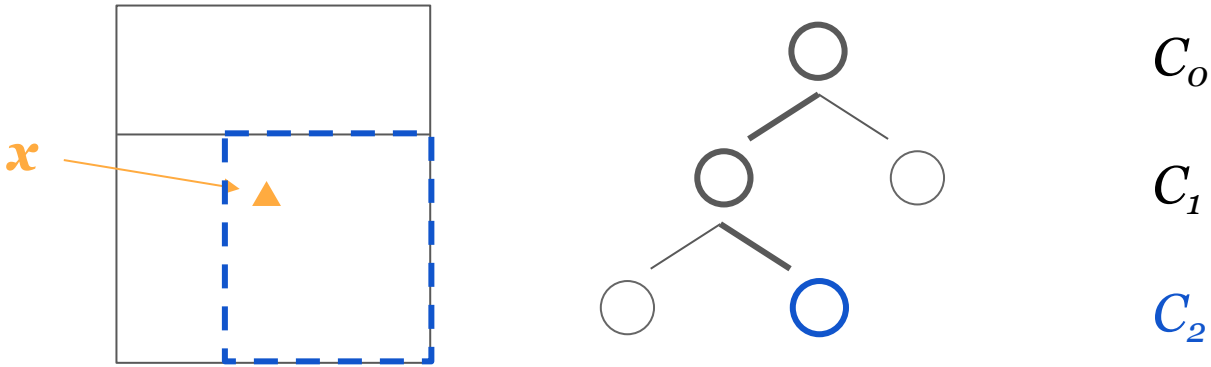




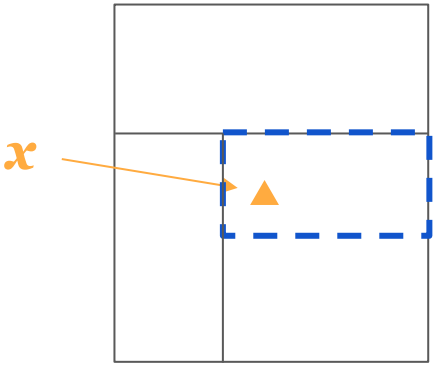
C_o

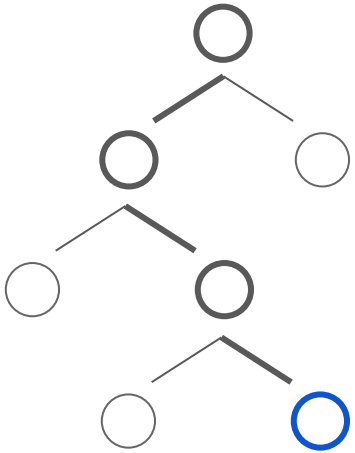
C_1

Root-to-leaf path is a stochastic process



Root-to-leaf path is a stochastic process





C_0

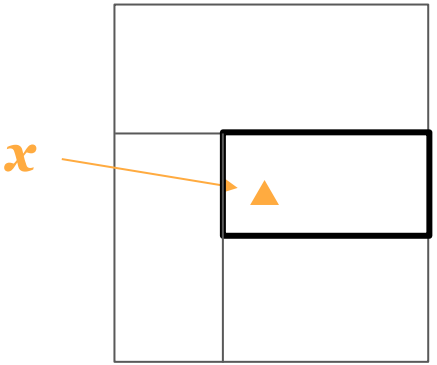
C_1

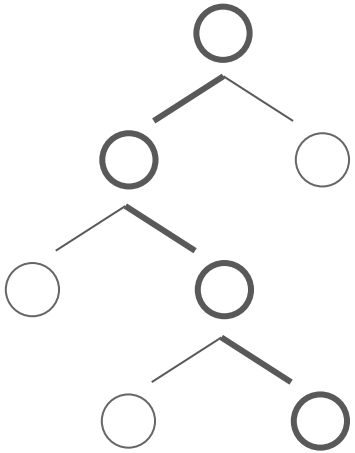
C_2

C_3

$(\hat{\Delta}_1, \ldots, \hat{\Delta}_d)$

Root-to-leaf path is a stochastic process





C_0

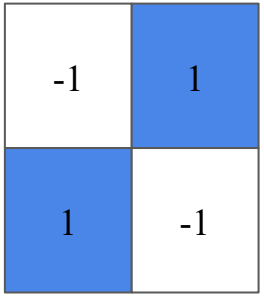
C_1

C_2

C_3

Use $J(\boldsymbol{x};\ \mathcal{D}_n) \subset [d]$ to denote the set of **features split upon** along the path

Reduction to bounding $\mathbb{P}\{1,2 \notin J(\boldsymbol{X}; \mathcal{D}_n)\}$



$$\mathbb{E}_{\mathbf{X},\mathcal{D}_n}\left\{\left(\hat{f}(\mathbf{X};\mathcal{D}_n) - f^*(\mathbf{X})\right)^2\right\}$$

tower property

$$= \mathbb{E}_{\mathbf{X},\mathcal{D}_n}\left\{\mathbb{E}_{\mathbf{X}}\left\{\left(\hat{f}(\mathbf{X};\mathcal{D}_n) - f^*(\mathbf{X})\right)^2 \mid \mathbf{X}_{[d]\setminus\{1,2\}}\right\}\right\}$$

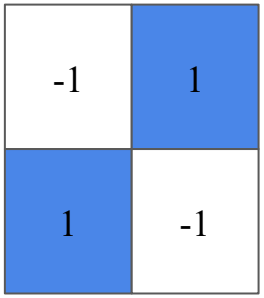
$$\geq \mathbb{E}_{\mathbf{X},\mathcal{D}_n}\left\{\mathbb{E}_{\mathbf{X}}\left\{\left(\hat{f}(\mathbf{X};\mathcal{D}_n) - f^*(\mathbf{X})\right)^2 \mid \mathbf{X}_{[d]\setminus\{1,2\}}\right\}\mathbf{1}\{1,2 \notin J(\mathbf{X};\mathcal{D}_n)\}\right\}$$

$$\geq \mathbb{E}_{\mathbf{X},\mathcal{D}_n}\left\{\mathrm{Var}\{f^*(\mathbf{X})\}\cdot\mathbf{1}\{1,2 \notin J(\mathbf{X};\mathcal{D}_n)\}\right\}$$

When $1,2 \notin J(\boldsymbol{X};\ \mathcal{D}_n)$, then

- $\hat{f}(\boldsymbol{X};\mathcal{D}_n)$ is constant with respect to $\boldsymbol{X}_{\{1,2\}}$
- $\mathbb{E}_X\left\{\left(\hat{f}(\boldsymbol{X};\mathcal{D}_n) - f^*(\boldsymbol{X})\right)^2 \mid \boldsymbol{X}_{[d]\setminus\{1,2\}}\right\} \geq \mathrm{Var}\{f^*(\boldsymbol{X})\}$

Reduction to bounding $\mathbb{P}\{1,2 \notin J(\boldsymbol{X}; \mathcal{D}_n)\}$



$$\mathbb{E}_{\mathbf{X},\mathcal{D}_n}\left\{\left(\hat{f}(\mathbf{X};\mathcal{D}_n) - f^*(\mathbf{X})\right)^2\right\}$$

tower property

$$= \mathbb{E}_{\mathbf{X},\mathcal{D}_n}\left\{\mathbb{E}_{\mathbf{X}}\left\{\left(\hat{f}(\mathbf{X};\mathcal{D}_n) - f^*(\mathbf{X})\right)^2 \mid \mathbf{X}_{[d]\setminus\{1,2\}}\right\}\right\}$$

$$\geq \mathbb{E}_{\mathbf{X},\mathcal{D}_n}\left\{\mathbb{E}_{\mathbf{X}}\left\{\left(\hat{f}(\mathbf{X};\mathcal{D}_n) - f^*(\mathbf{X})\right)^2 \mid \mathbf{X}_{[d]\setminus\{1,2\}}\right\}\mathbf{1}\{1,2 \notin J(\mathbf{X};\mathcal{D}_n)\}\right\}$$

$$\geq \mathbb{E}_{\mathbf{X},\mathcal{D}_n}\left\{\mathrm{Var}\{f^*(\mathbf{X})\}\cdot\mathbf{1}\{1,2 \notin J(\mathbf{X};\mathcal{D}_n)\}\right\}$$

$$= \mathrm{Var}\{f^*(\mathbf{X})\}\cdot\mathbb{P}\{1,2 \notin J(\mathbf{X};\mathcal{D}_n)\}$$

Bounding $\mathbb{P}\{1,2 \notin J(\boldsymbol{X}; \mathcal{D}_n)\}$ via **coupling**

- Want to say that the root-to-leaf path splits on a random subset of features

C_0

C_1

C_2

C_3

$\vdots$

Bounding $\mathbb{P}\{1,2 \notin J(\boldsymbol{X}; \mathcal{D}_n)\}$ via **coupling**

- Want to say that the root-to-leaf path splits on a random subset of features
- Hard to prove directly, so construct a related path

C_0 $C_0^{(1)}$

C_1 $C_1^{(1)}$

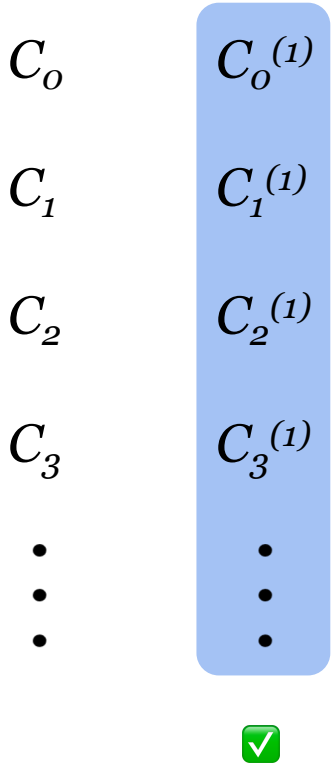
C_2 $C_2^{(1)}$

C_3 $C_3^{(1)}$

⋮ ⋮

Bounding $\mathbb{P}\{1,2 \notin J(\boldsymbol{X}; \mathcal{D}_n)\}$ via **coupling**

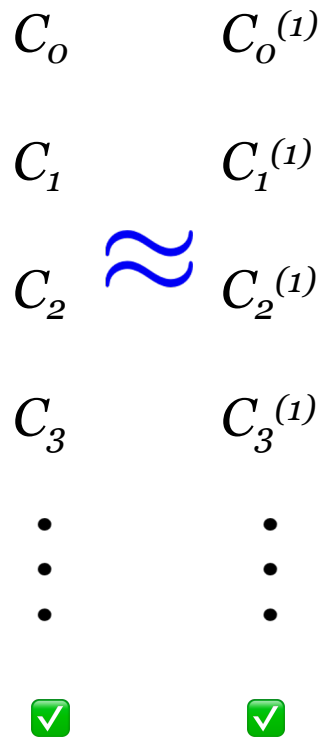
- Want to say that the root-to-leaf path splits on a random subset of features
- Hard to prove directly, so construct a related path
- Prove the desired property for the related path



✅

Bounding $\mathbb{P}\{1,2 \notin J(\boldsymbol{X}; \mathcal{D}_n)\}$ via **coupling**

- Want to say that the root-to-leaf path splits on a random subset of features
- Hard to prove directly, so construct a related path
- Prove the desired property for the related path



Key Takeaways

- Explored the limits of greedy optimization with CART & RFs
- Some functions (e.g., XOR) are hard to learn
- Lower bounds reveal fundamental **statistical-computational** trade-offs
- CART & RFs vs. NNs. When to use each method?
- Tan, K., Balasubramanian, *Statistical-Computational Trade-offs for Recursive Adaptive Partitioning Estimators*, Major revision in AoS, 2024+

Thank you!

Questions?

jason.klusowski@princeton.edu