BART as a Gaussian process

Giacomo Petrillo

2023-02-02

Department of Statistics, Computer Science, Applications (DISIA)
University of Florence

BART

- BART = Bayesian Additive Regression Trees (Chipman et al. 2010)

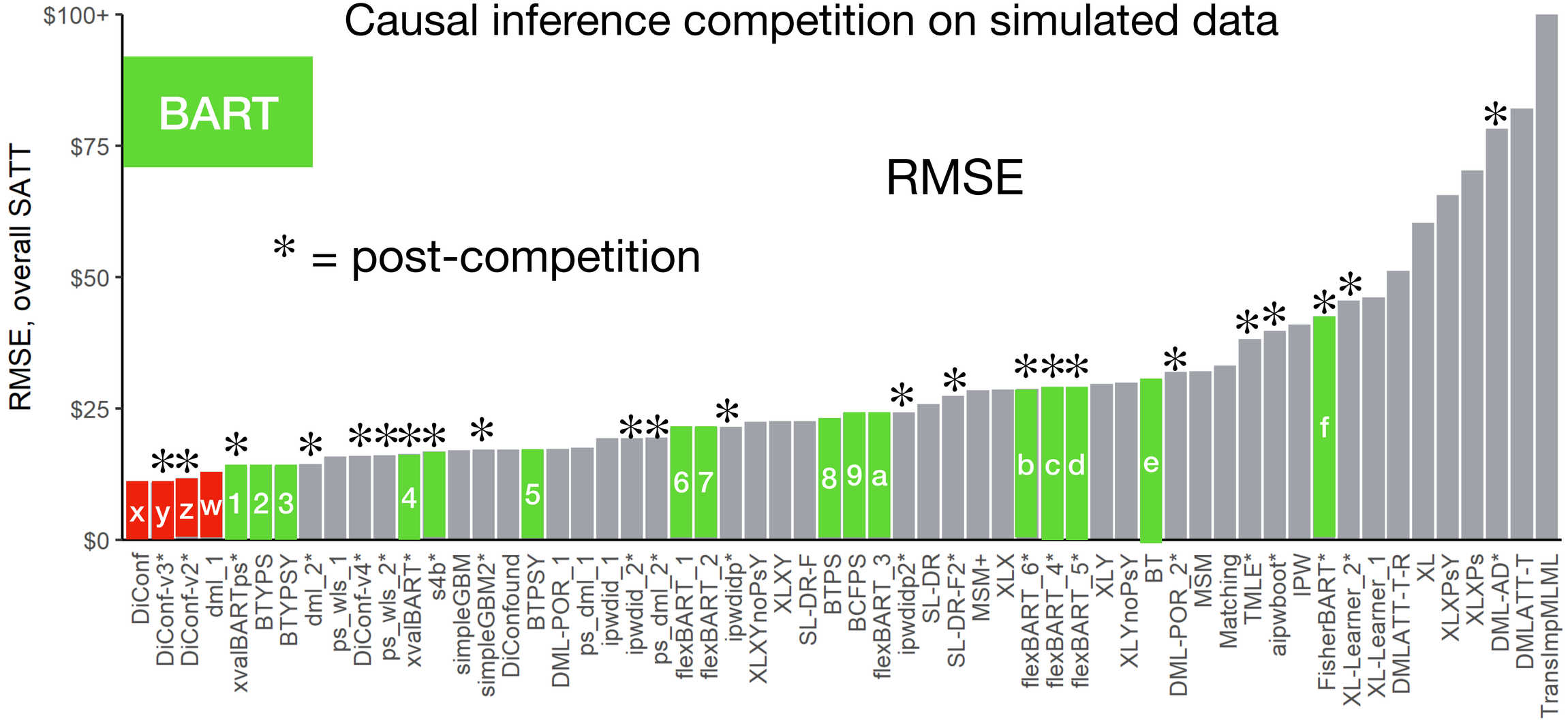
BART

- BART = Bayesian Additive Regression Trees (Chipman et al. 2010)
- Almost fully Bayesian nonparametric regression

BART

- BART = Bayesian Additive Regression Trees (Chipman et al. 2010)
- Almost fully Bayesian nonparametric regression
- Star in causal inference

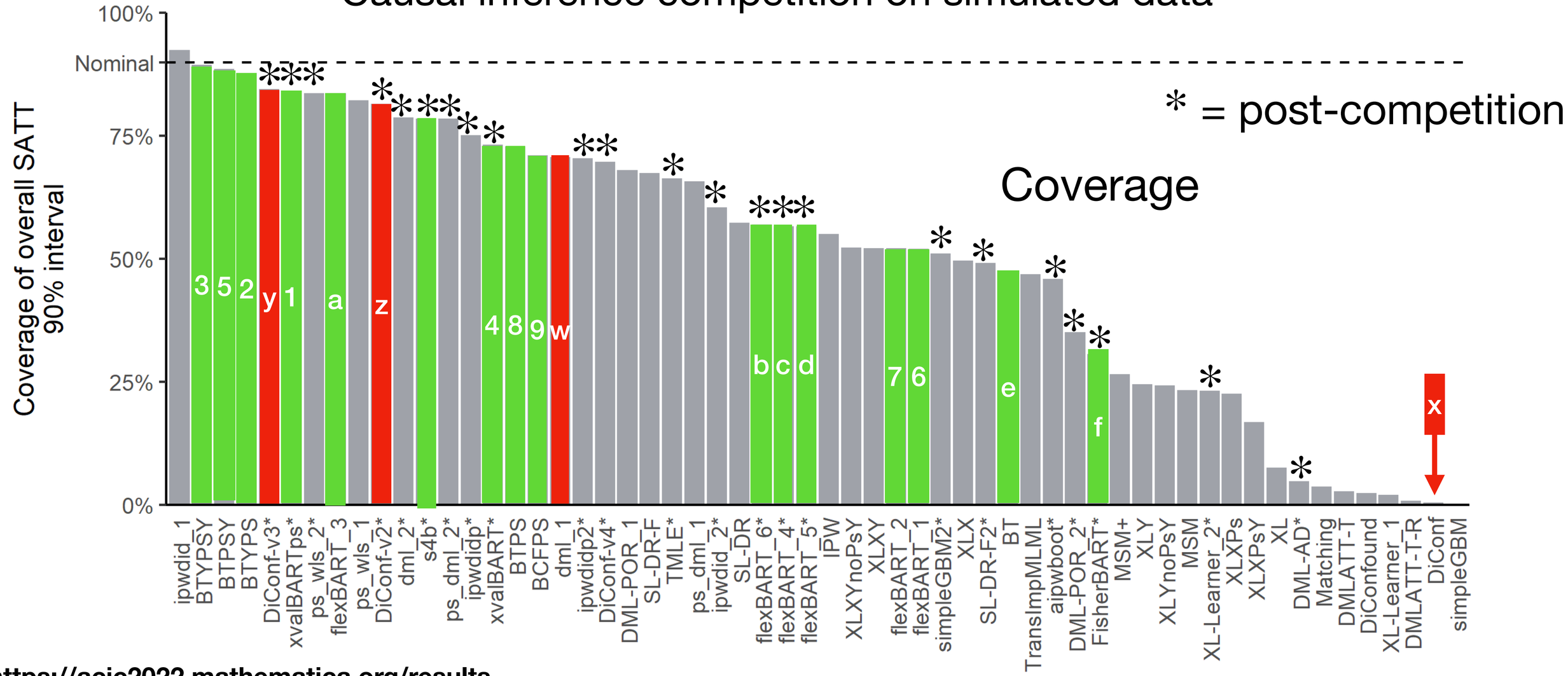
ACIC Data Challenge



https://acic2022.mathematica.org/results

ACIC Data Challenge

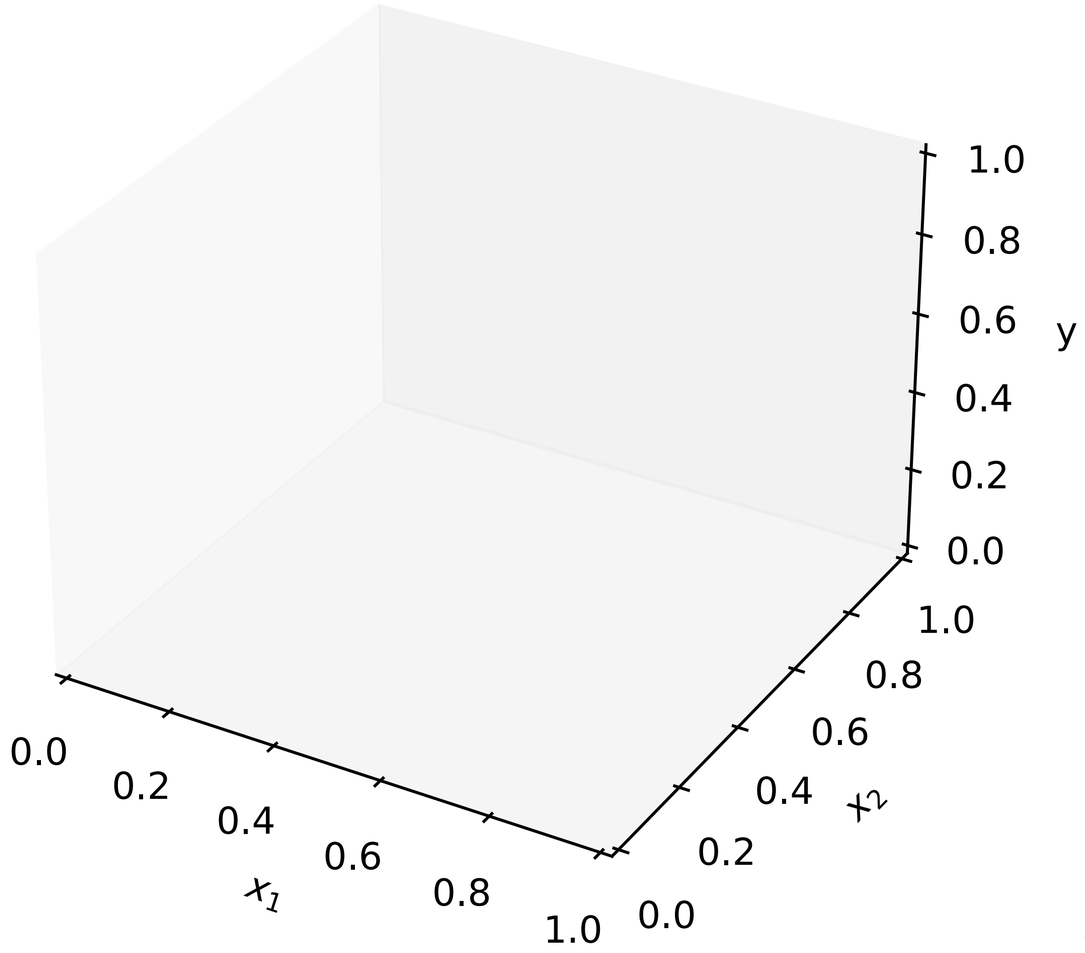
Causal inference competition on simulated data



https://acic2022.mathematica.org/results

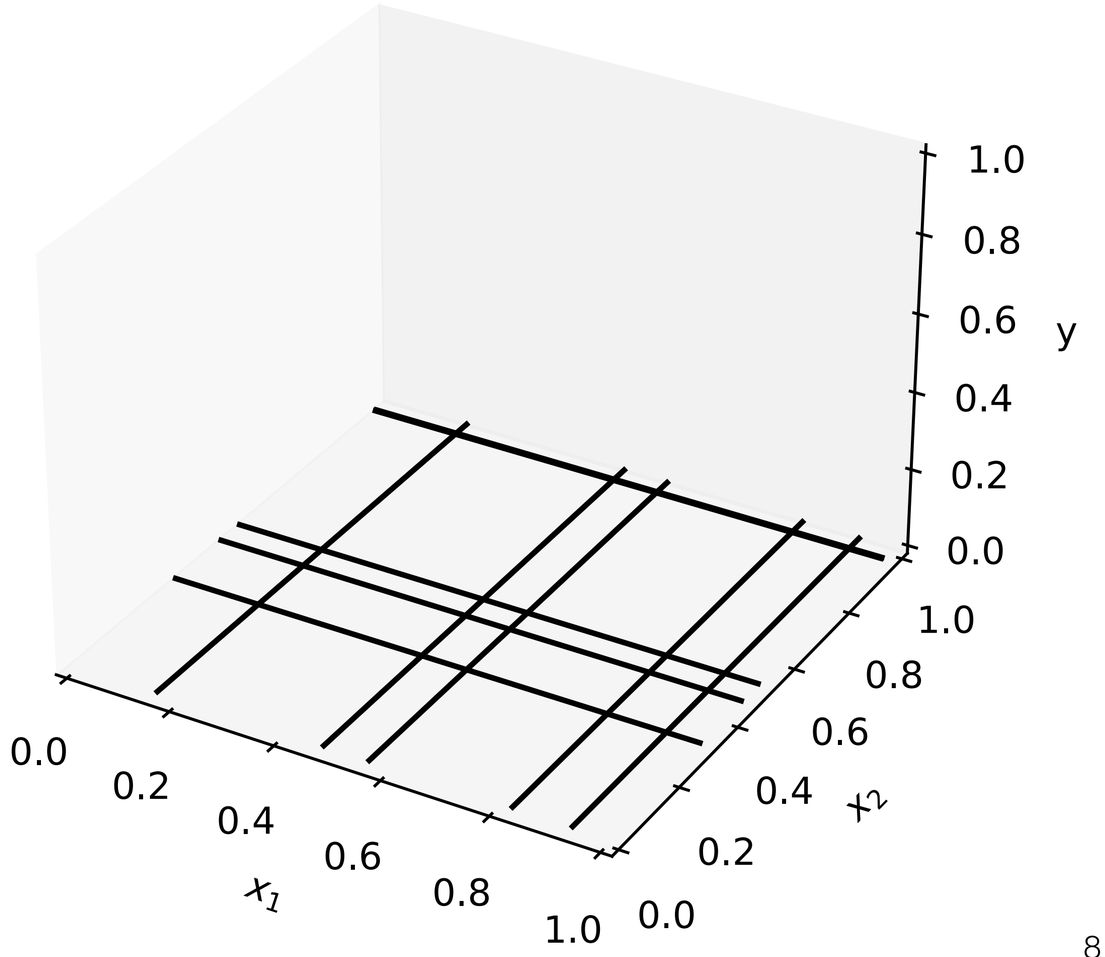
Definition of BART

- Start from a single regression tree $g(x; T, M)$



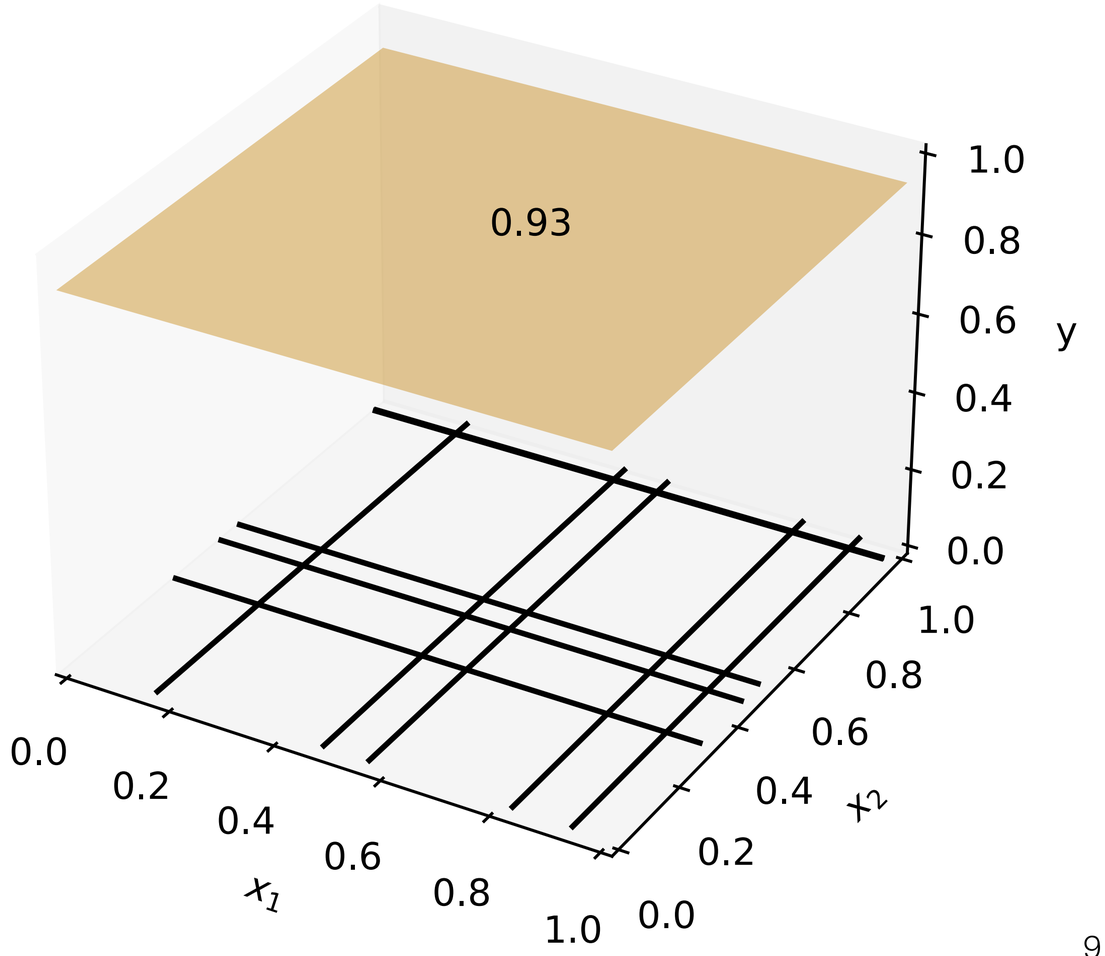
Definition of BART

- Start from a single regression tree $g(x; T, M)$



Definition of BART

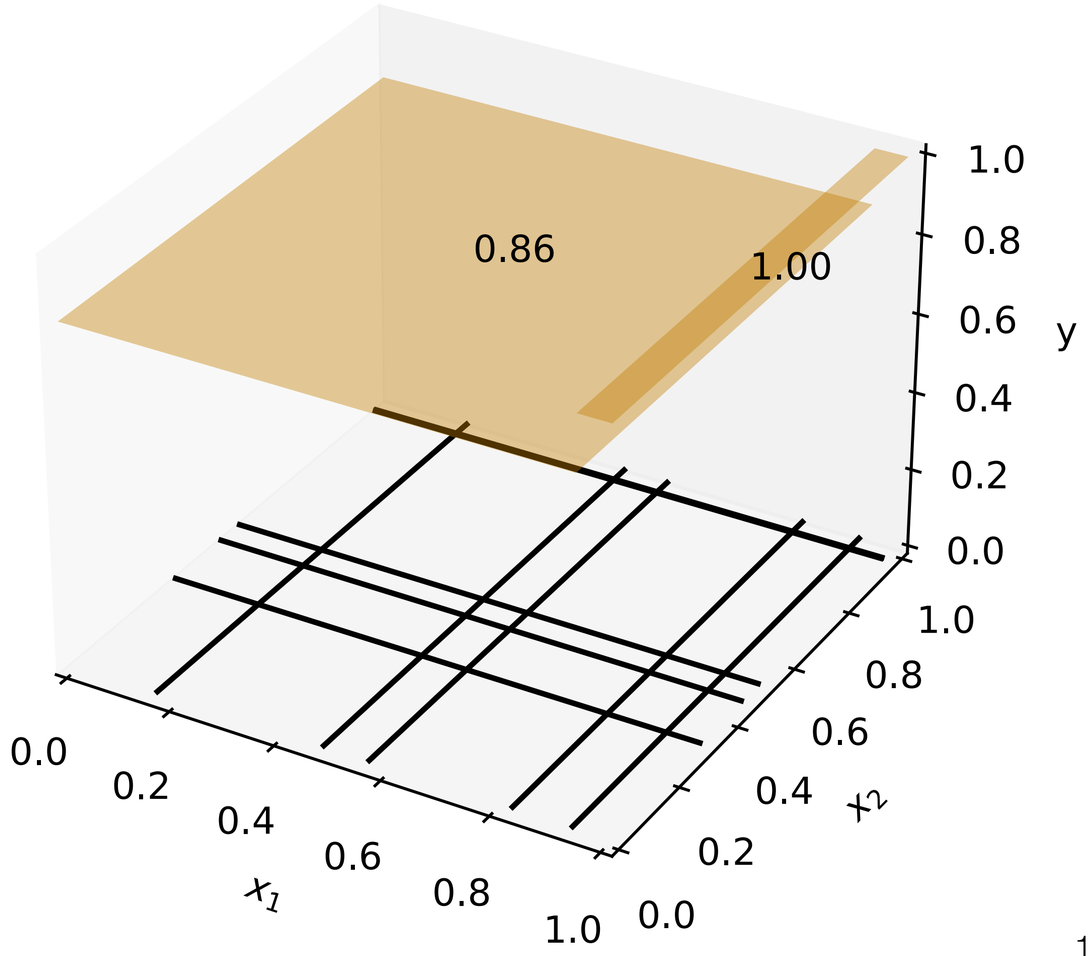
- Start from a single regression tree $g(x; T, M)$

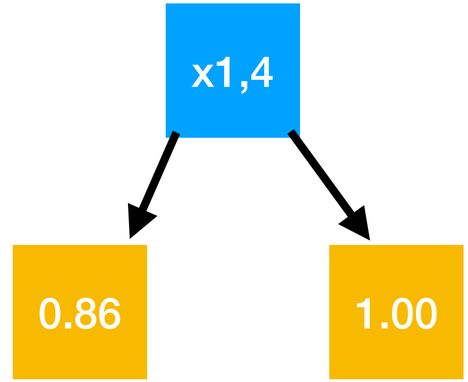


0.93

Definition of BART

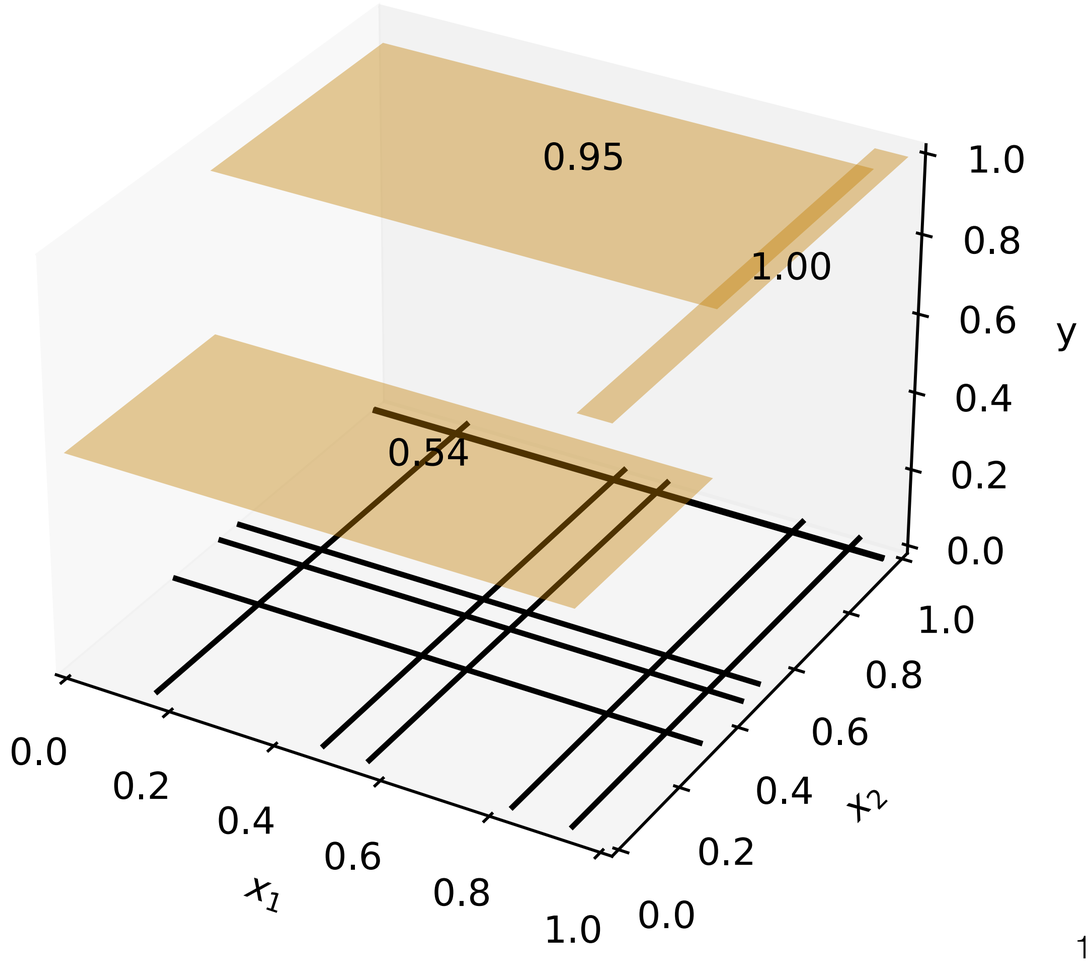
- Start from a single regression tree $g(x; T, M)$

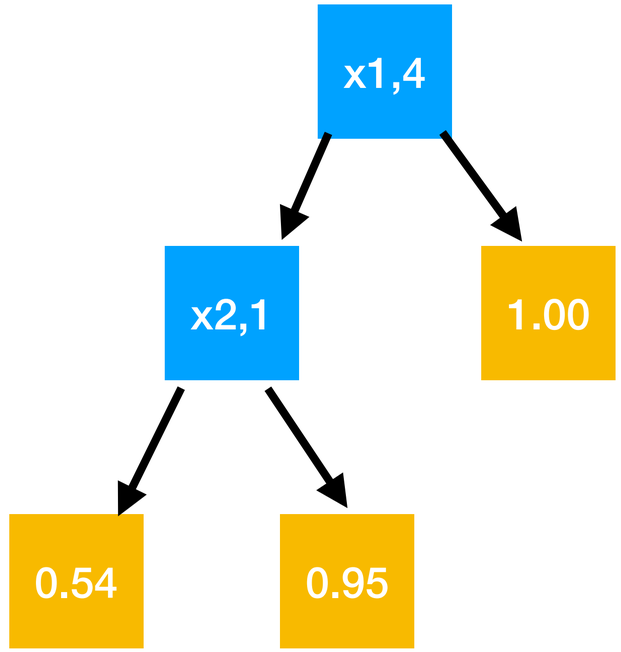




Definition of BART

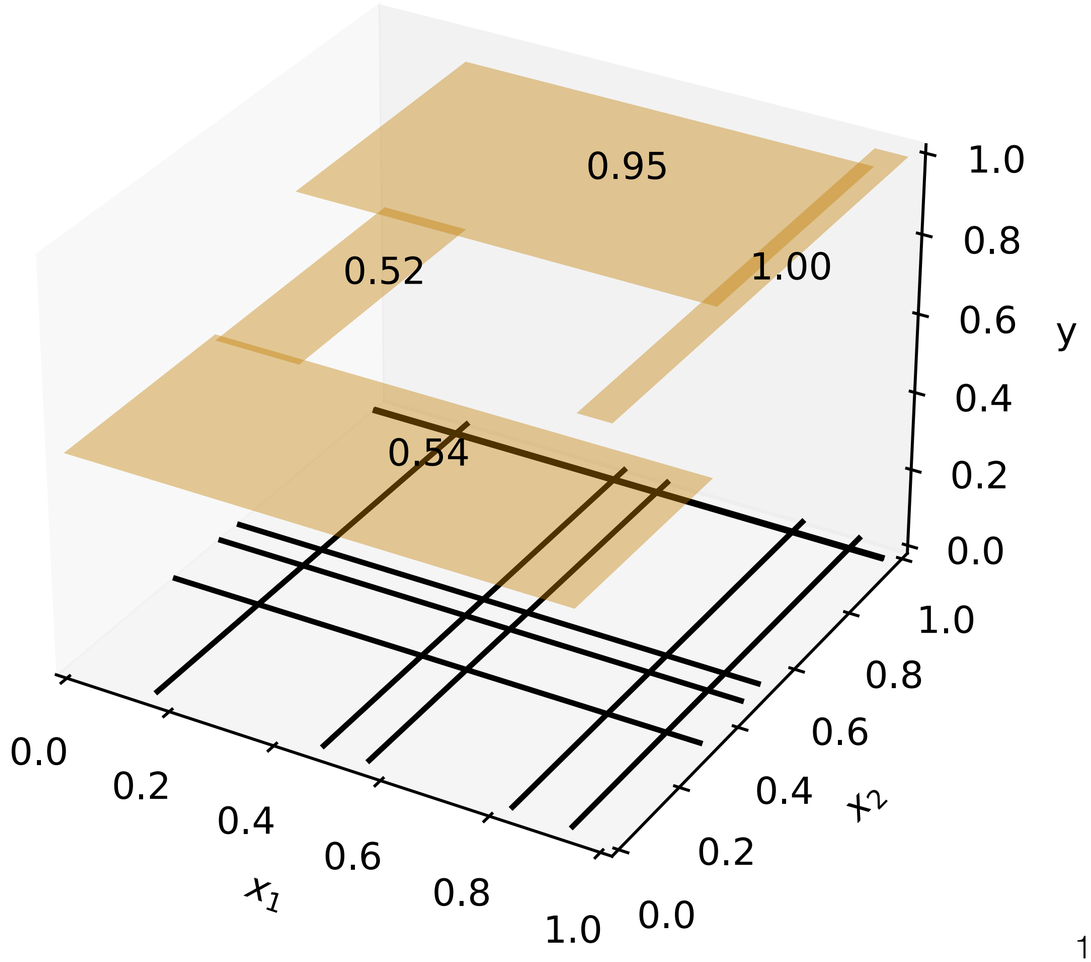
- Start from a single regression tree $g(x; T, M)$

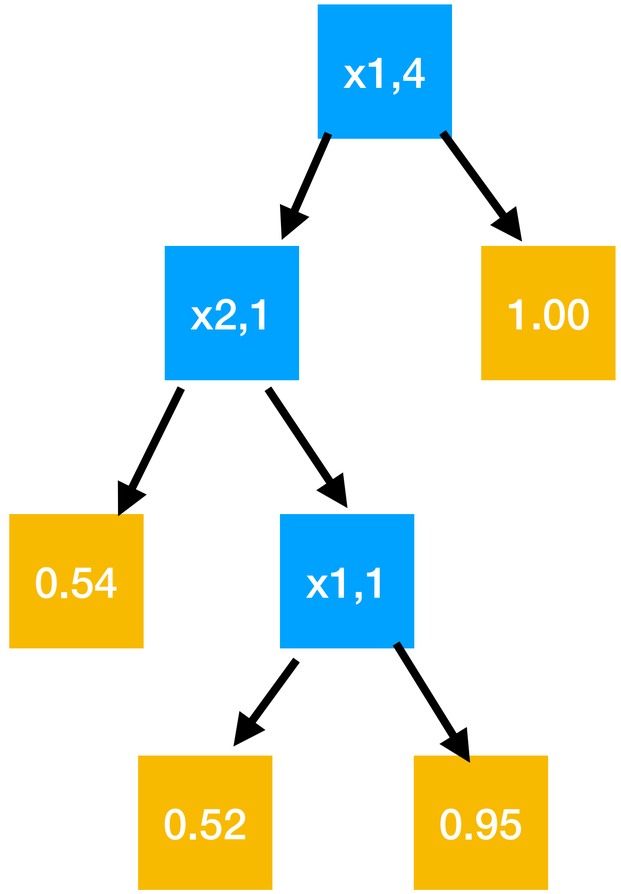




Definition of BART

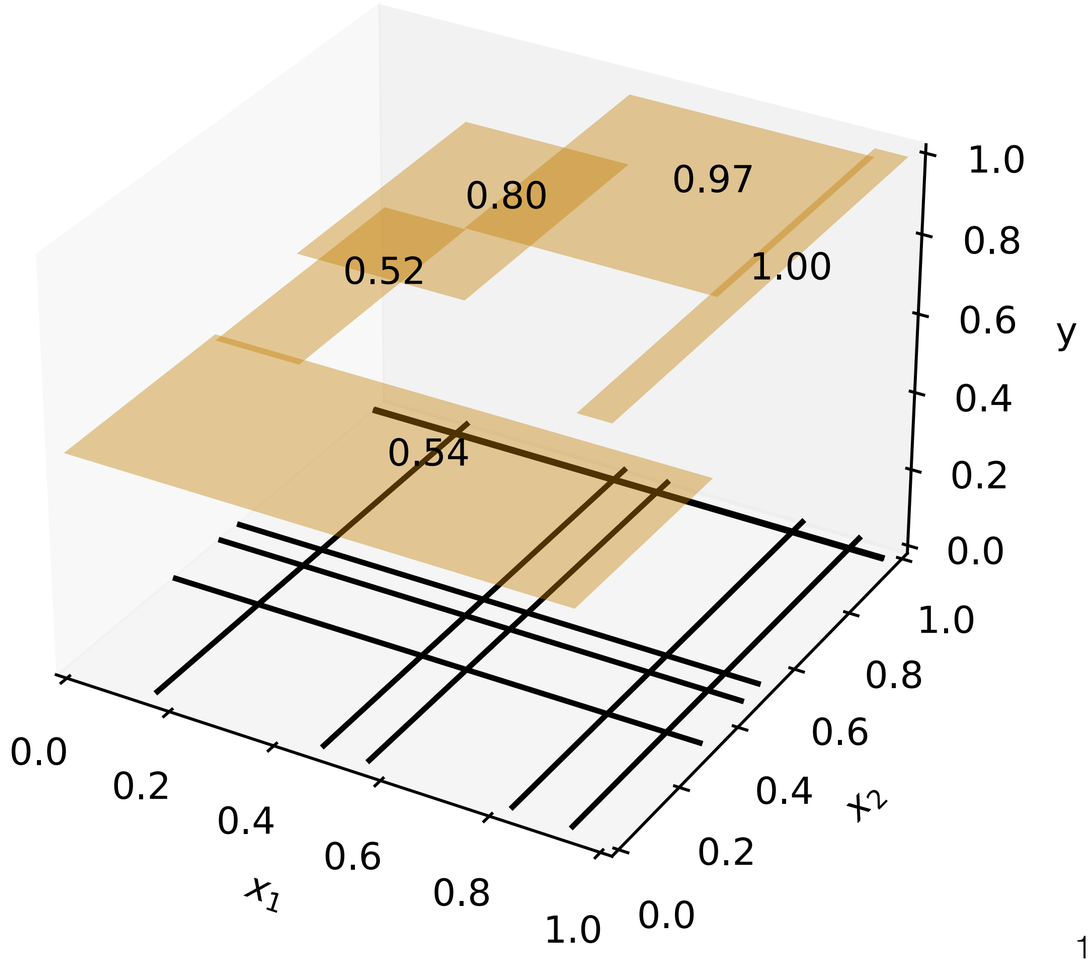
- Start from a single regression tree $g(x; T, M)$

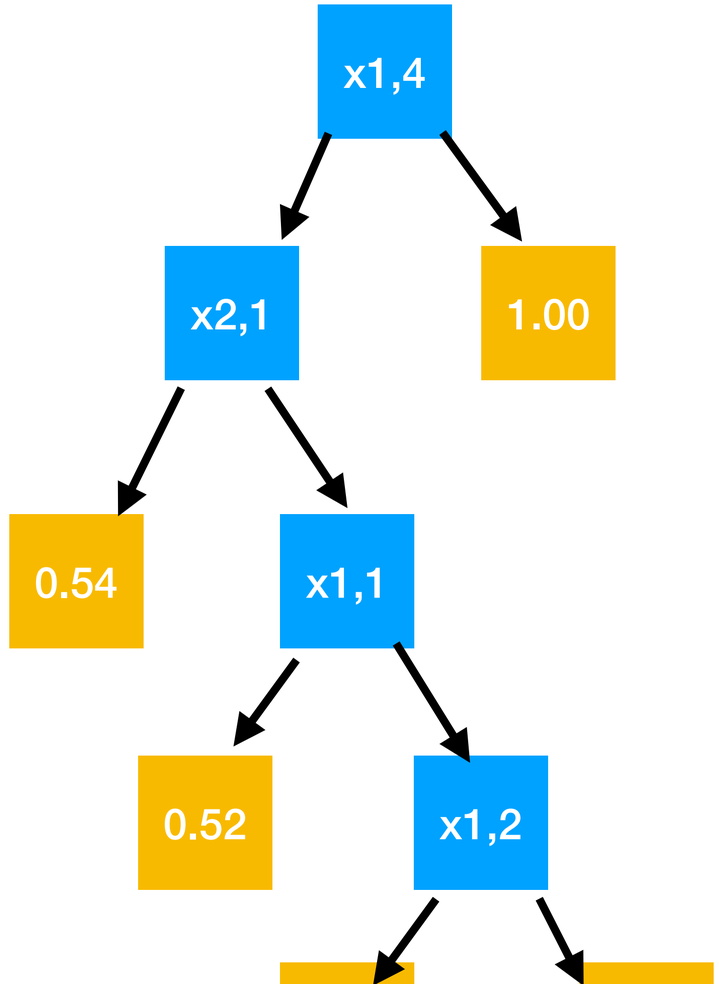




Definition of BART

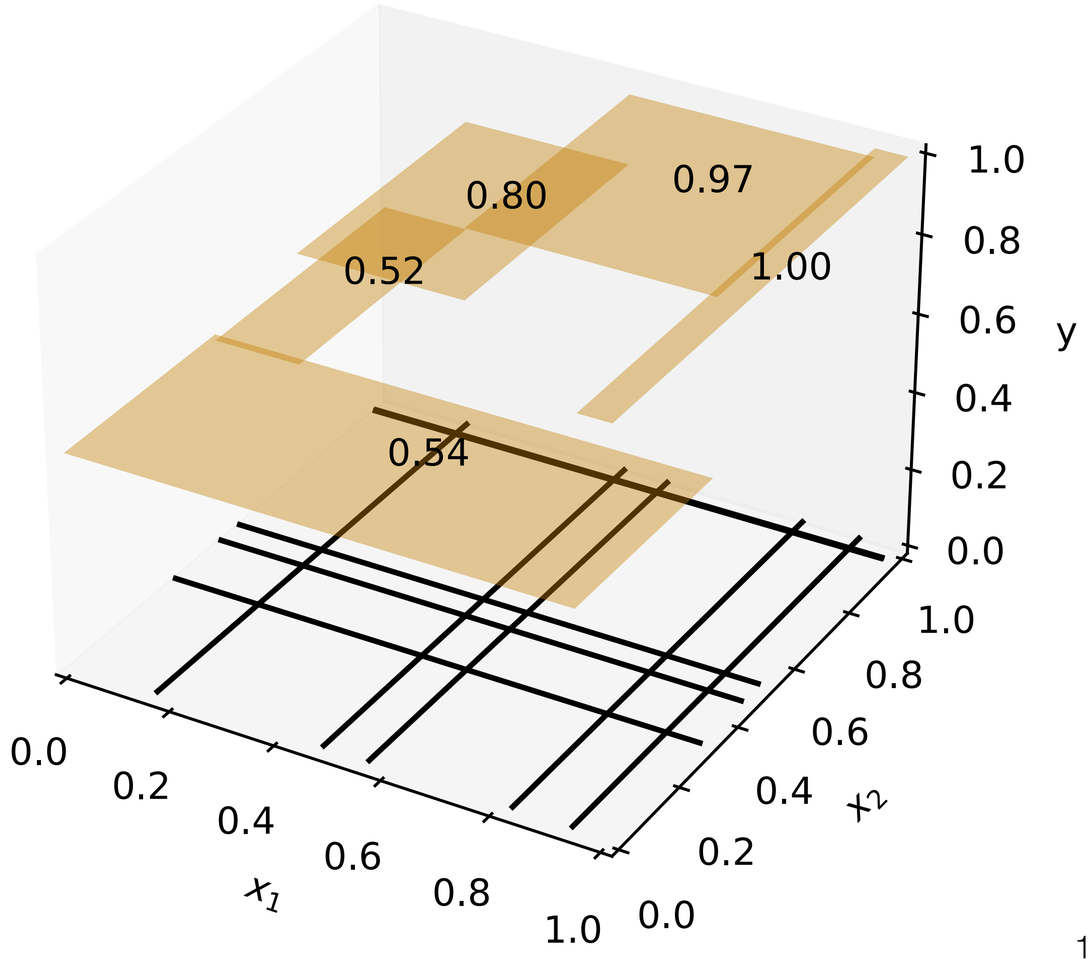
- Start from a single regression tree $g(x; T, M)$

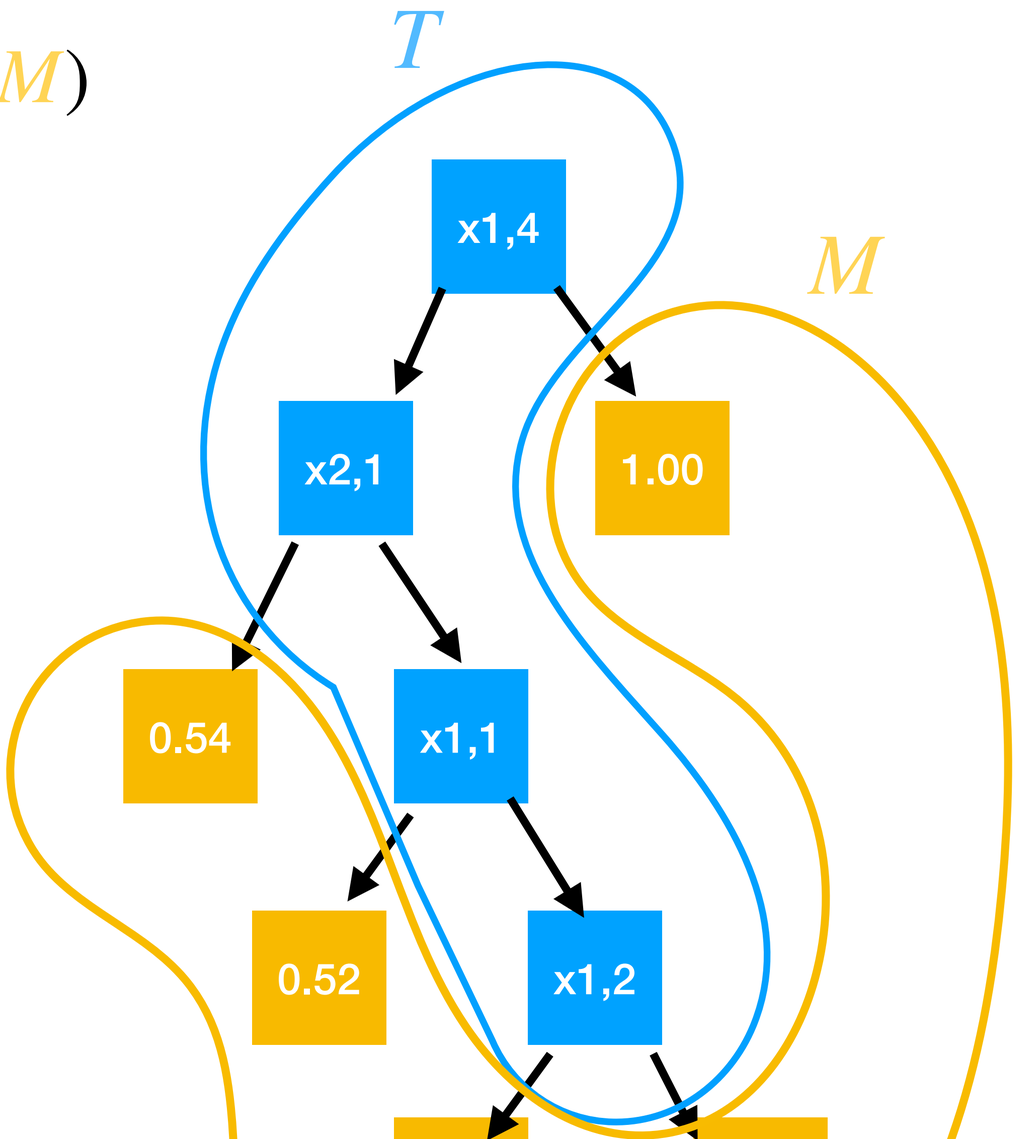




Definition of BART

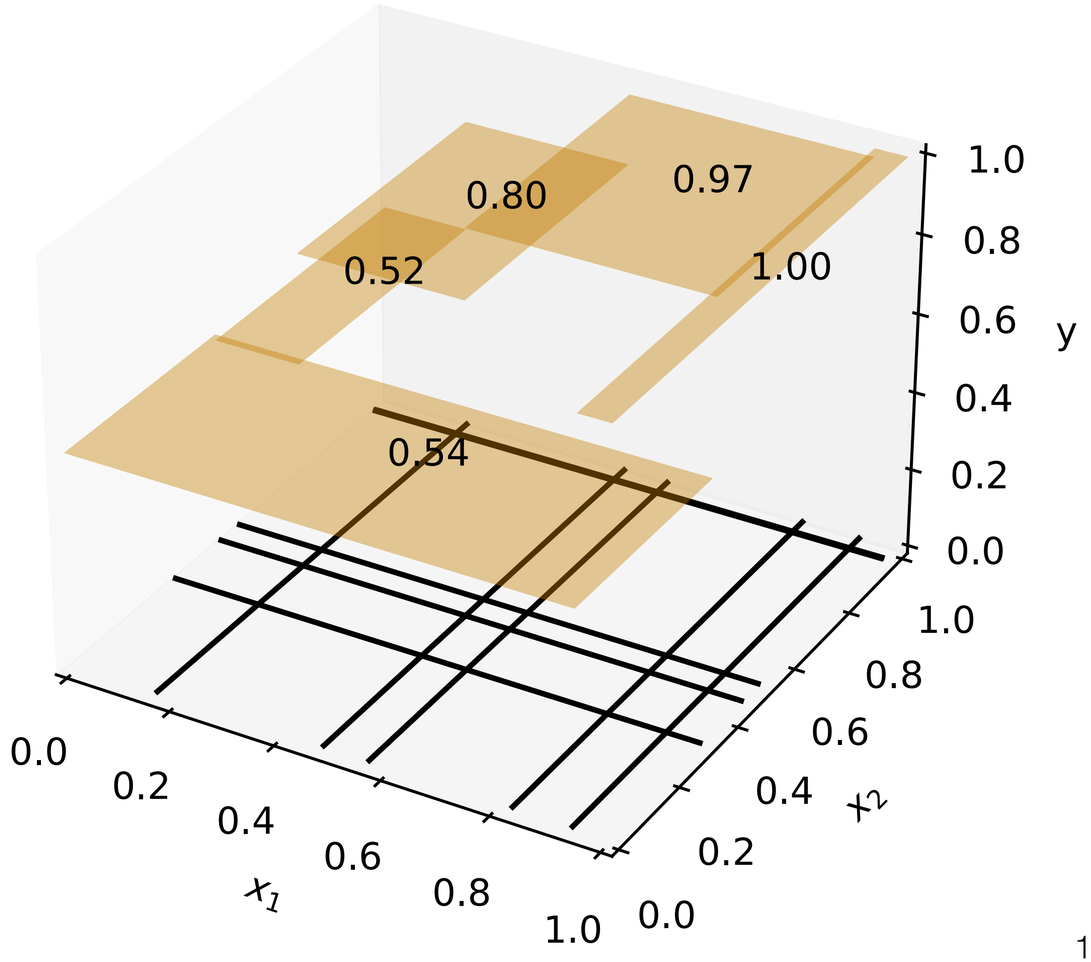
- Start from a single regression tree $g(x; T, M)$



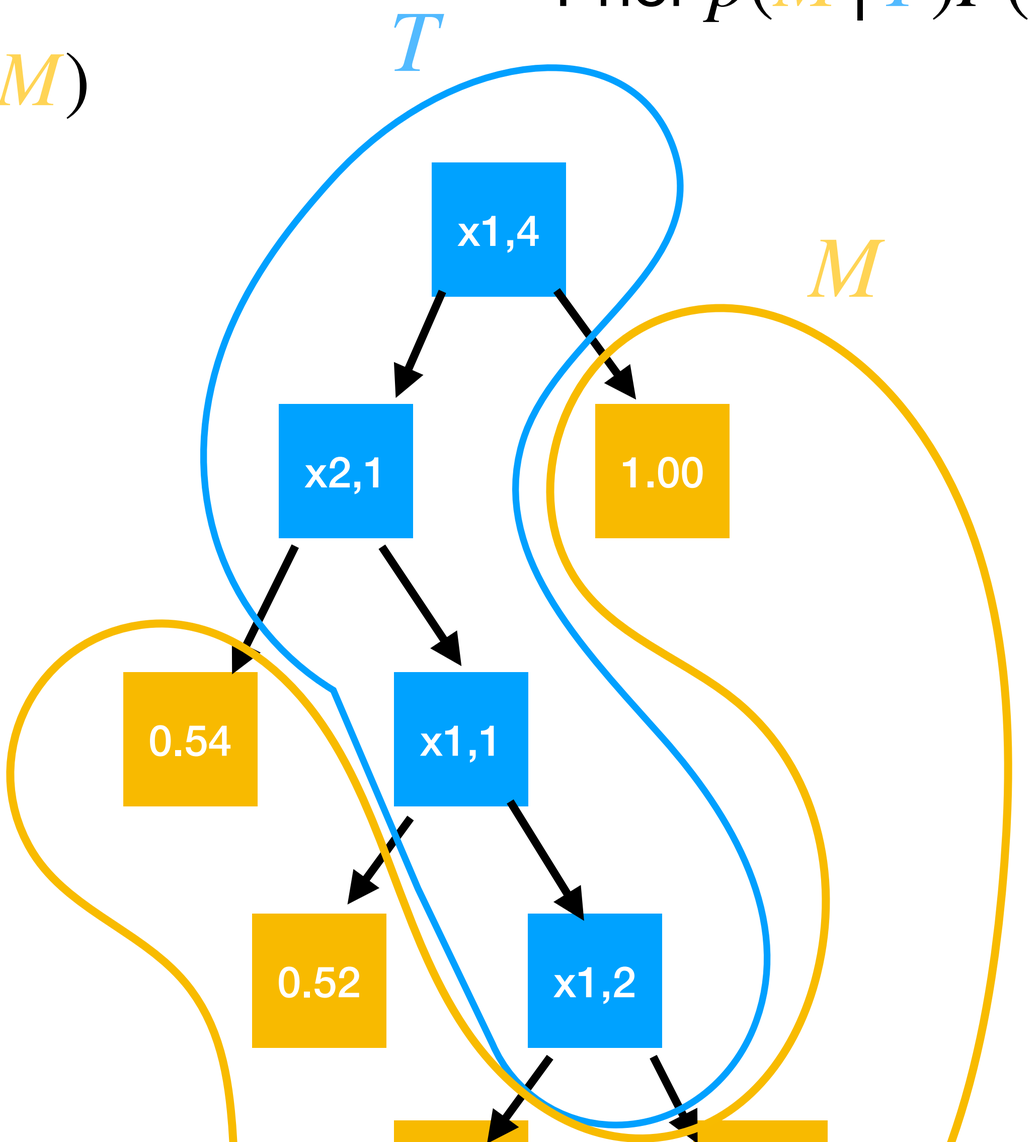


Definition of BART

- Start from a single regression tree $g(x; T, M)$



Prior $p(M \mid T)P(T)$



Definition of BART—more trees

- Consider m a priori i.i.d. regression trees $g(x; T_j, M_j)$

Definition of BART—more trees

- Consider m a priori i.i.d. regression trees $g(x; T_j, M_j)$
- $$y = \sum_{j=1}^{m} g(x; T_j, M_j) + \varepsilon(x)$$

Definition of BART—more trees

- Consider m a priori i.i.d. regression trees $g(x; T_j, M_j)$
- $$y = \sum_{j=1}^{m} g(x; T_j, M_j) + \varepsilon(x)$$
- $$\varepsilon(x) \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2)$$

Definition of BART—more trees

- Consider m a priori i.i.d. regression trees $g(x; T_j, M_j)$
- $$y = \sum_{j=1}^{m} g(x; T_j, M_j) + \varepsilon(x)$$
- $\varepsilon(x) \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2)$
- $T_j, M_j \overset{\text{i.i.d.}}{\sim} p(M_j \mid T_j) P(T_j)$

Definition of BART—more trees

- Consider m a priori i.i.d. regression trees $g(x; T_j, M_j)$
- $$y = \sum_{j=1}^{m} g(x; T_j, M_j) + \varepsilon(x)$$
- $\varepsilon(x) \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2)$
- $T_j, M_j \overset{\text{i.i.d.}}{\sim} p(M_j \mid T_j) P(T_j)$
- The tree prior depends on hyperparameters α, β that tune depth and balance

BART — Inference

- Metropolis-Gibbs sampler, one tree at a time

BART — Inference

- Metropolis-Gibbs sampler, one tree at a time
- Why many trees?

BART — Inference

- Metropolis-Gibbs sampler, one tree at a time
- Why many trees?
- One large and deep tree is difficult to update with Metropolis

BART — Inference

- Metropolis-Gibbs sampler, one tree at a time
- Why many trees?
- One large and deep tree is difficult to update with Metropolis
- Does this get stuck?

BART — Inference

- Metropolis-Gibbs sampler, one tree at a time
- Why many trees?
- One large and deep tree is difficult to update with Metropolis
- Does this get stuck?
- The error term leaves room for a single tree to change (inverse-gamma on σ^2)

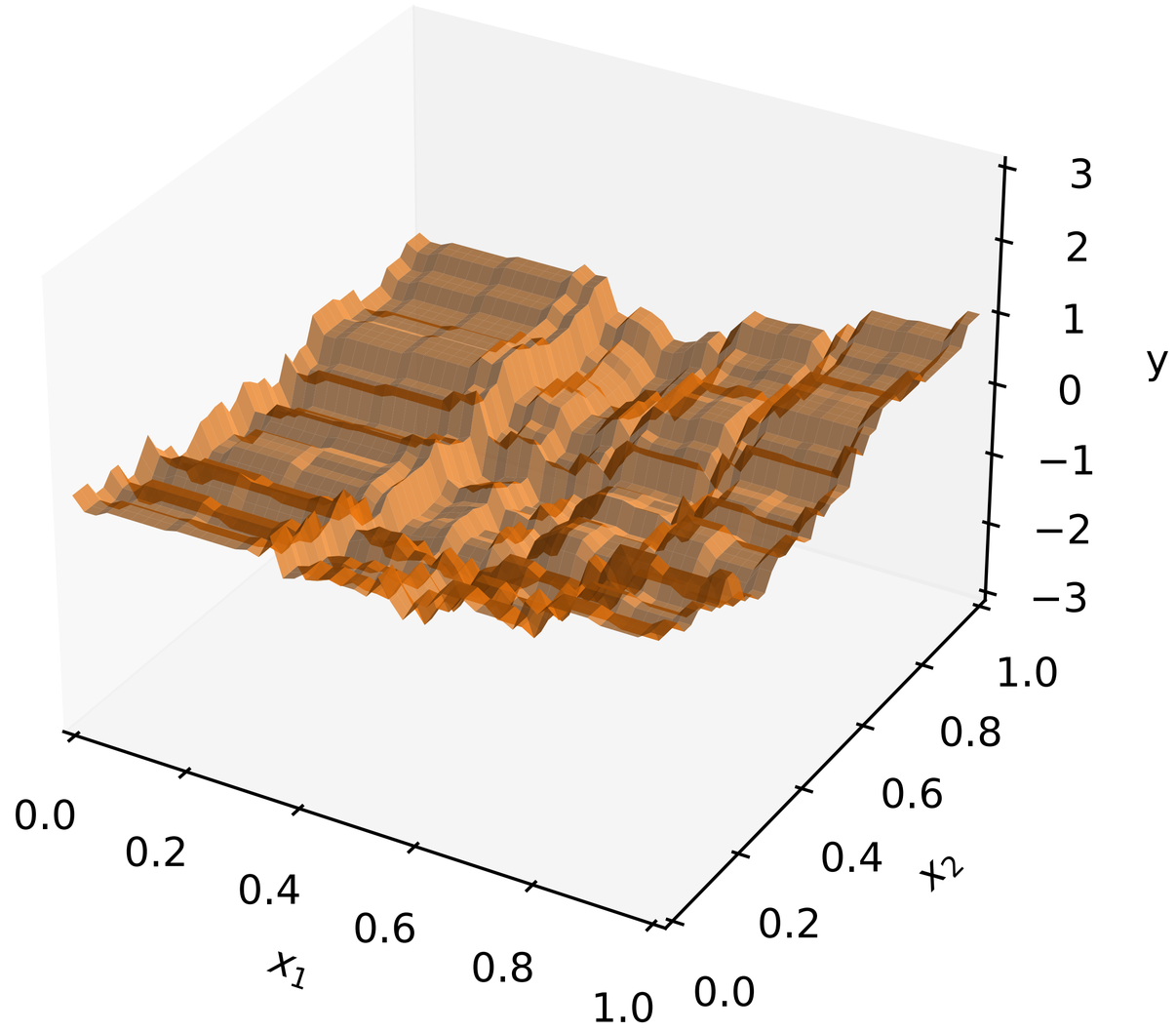
BART — Inference

- Metropolis-Gibbs sampler, one tree at a time
- Why many trees?
- One large and deep tree is difficult to update with Metropolis
- Does this get stuck?
- The error term leaves room for a single tree to change (inverse-gamma on σ^2)
- $m \approx 200$

(This slide contains speculations)

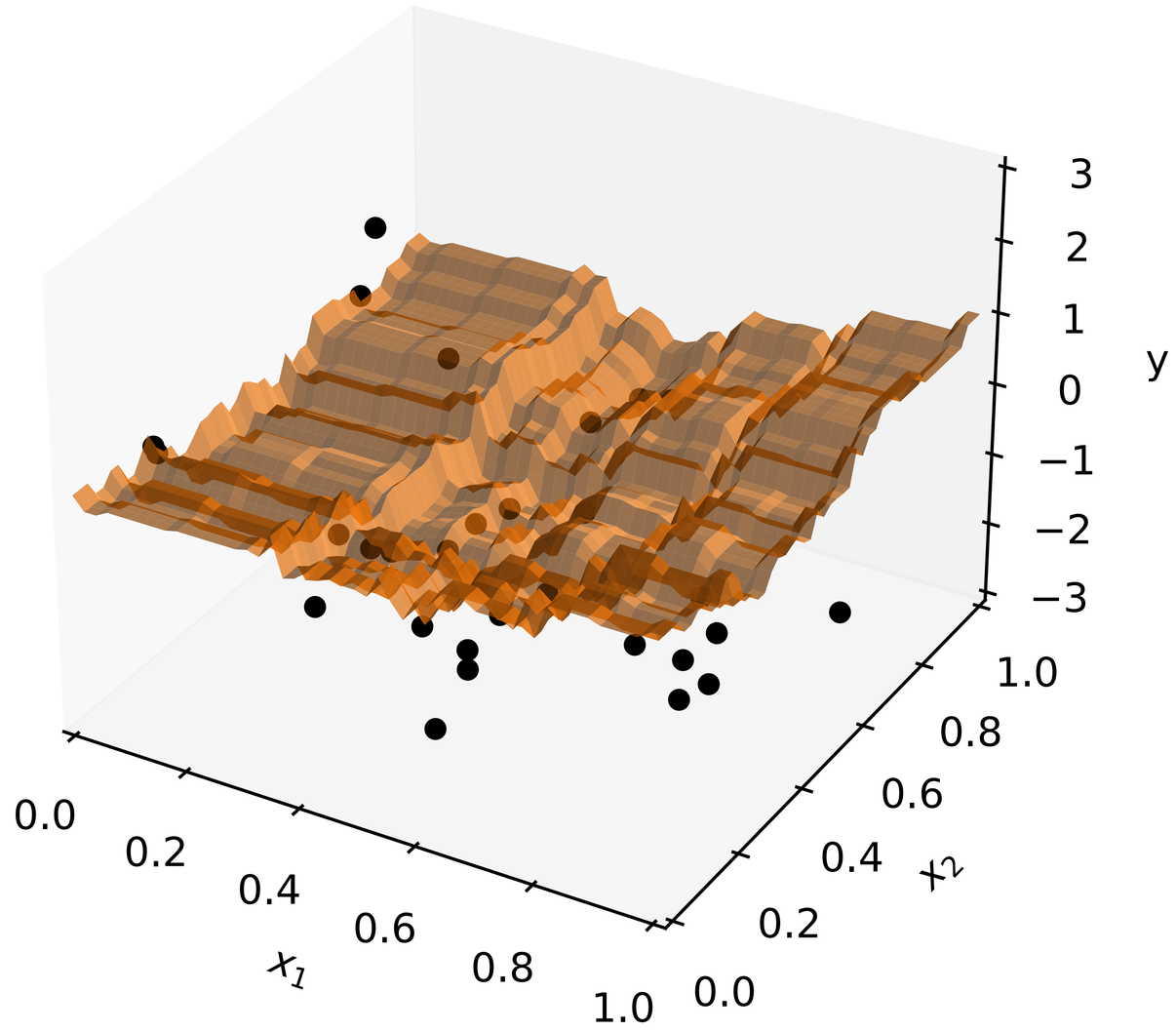
Example

Prior sample



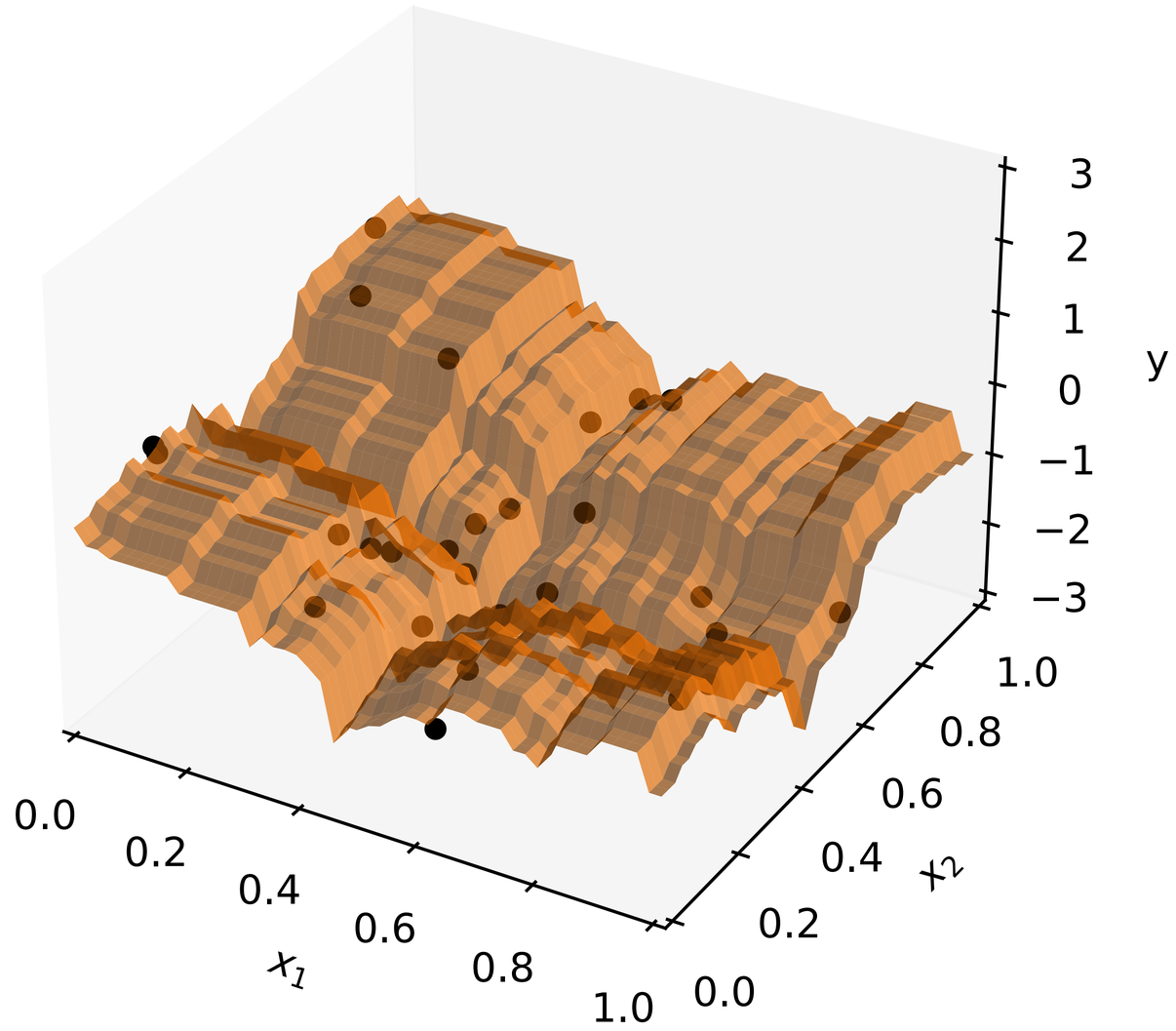
Example

Data



Example

Posterior sample



Gaussian process regression

- Another Bayesian nonparametric method

Gaussian process regression

- Another Bayesian nonparametric method
- Gaussian process = Multivariate Normal in ∞ dimensions

Gaussian process regression

- Another Bayesian nonparametric method
- Gaussian process = Multivariate Normal in ∞ dimensions
- Finite marginals are Normal

Gaussian process regression

- Another Bayesian nonparametric method
- Gaussian process = Multivariate Normal in ∞ dimensions
- Finite marginals are Normal
- $\Longrightarrow$ Analytical calculations

Gaussian process regression

- Another Bayesian nonparametric method
- Gaussian process = Multivariate Normal in ∞ dimensions
- Finite marginals are Normal
- $\Longrightarrow$ Analytical calculations
- A priori $$\begin{pmatrix} f(\mathbf{x}_1) \\ \vdots \\ f(\mathbf{x}_n) \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} m(\mathbf{x}_1) \\ \vdots \\ m(\mathbf{x}_n) \end{pmatrix}, \begin{pmatrix} k(\mathbf{x}_1, \mathbf{x}_1) & \cdots & k(\mathbf{x}_1, \mathbf{x}_n) \\ \vdots & \ddots & \vdots \\ k(\mathbf{x}_n, \mathbf{x}_1) & \cdots & k(\mathbf{x}_n, \mathbf{x}_n) \end{pmatrix} \right)$$

GP — Inference

- We know $(f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n)) = \mathbf{y}$

GP — Inference

- We know $(f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n)) = \mathbf{y}$
- We want $f(\mathbf{x}_1^*), \ldots, f(\mathbf{x}_m^*)$

GP — Inference

- We know $(f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n)) = \mathbf{y}$
- We want $f(\mathbf{x}_1^*), \ldots, f(\mathbf{x}_m^*)$
- $$\begin{pmatrix} f(\mathbf{x}_1) \\ \vdots \\ f(\mathbf{x}_n) \\ f(\mathbf{x}_1^*) \\ \vdots \\ f(\mathbf{x}_m^*) \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} m(\mathbf{x}_1) \\ \vdots \\ m(\mathbf{x}_n) \\ m(\mathbf{x}_1^*) \\ \vdots \\ m(\mathbf{x}_m^*) \end{pmatrix}, \begin{pmatrix} k(\mathbf{x}_1, \mathbf{x}_1) & \cdots & k(\mathbf{x}_1, \mathbf{x}_n) & k(\mathbf{x}_1, \mathbf{x}_1^*) & \cdots & k(\mathbf{x}_1, \mathbf{x}_m^*) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ k(\mathbf{x}_n, \mathbf{x}_1) & \cdots & k(\mathbf{x}_n, \mathbf{x}_n) & k(\mathbf{x}_n, \mathbf{x}_1^*) & \cdots & k(\mathbf{x}_n, \mathbf{x}_m^*) \\ k(\mathbf{x}_1^*, \mathbf{x}_1) & \cdots & k(\mathbf{x}_1^*, \mathbf{x}_n) & k(\mathbf{x}_1^*, \mathbf{x}_1^*) & \cdots & k(\mathbf{x}_1^*, \mathbf{x}_m^*) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ k(\mathbf{x}_m^*, \mathbf{x}_1) & \cdots & k(\mathbf{x}_m^*, \mathbf{x}_n) & k(\mathbf{x}_m^*, \mathbf{x}_1^*) & \cdots & k(\mathbf{x}_m^*, \mathbf{x}_m^*) \end{pmatrix} \right)$$

GP — Inference

- Abbreviate $\mathbf{f} = (f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n)), \quad \mathbf{f}^* = (f(\mathbf{x}_1^*), \ldots, f(\mathbf{x}_m^*))$

GP — Inference

- Abbreviate $\mathbf{f} = (f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n)), \quad \mathbf{f}^* = (f(\mathbf{x}_1^*), \ldots, f(\mathbf{x}_m^*))$
- Abbreviate $\begin{pmatrix} \mathbf{f} \\ \mathbf{f}^* \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \mathbf{m} \\ \mathbf{m}^* \end{pmatrix}, \begin{pmatrix} \Sigma_{xx} & \Sigma_{xx^*} \\ \Sigma_{x^*x} & \Sigma_{x^*x^*} \end{pmatrix} \right)$

GP — Inference

- Abbreviate $\mathbf{f} = (f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n)), \quad \mathbf{f}* = (f(\mathbf{x}_1^*), \ldots, f(\mathbf{x}_m^*))$
- Abbreviate $\begin{pmatrix} \mathbf{f} \\ \mathbf{f}* \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \mathbf{m} \\ \mathbf{m}* \end{pmatrix}, \begin{pmatrix} \Sigma_{xx} & \Sigma_{xx*} \\ \Sigma_{x*x} & \Sigma_{x*x*} \end{pmatrix} \right)$
- $(\mathbf{f}* \mid \mathbf{f} = \mathbf{y}) \sim \mathcal{N}(\mathbf{m}* + \Sigma_{x*x}\Sigma_{xx}^{+}(\mathbf{y} - \mathbf{m}), \quad \Sigma_{x*x*} - \Sigma_{x*x}\Sigma_{xx}^{+}\Sigma_{xx*})$

BART ≠ GP

- “Given its underlying tree structure, intuitively **BART may not have the flexibility** to capture the additional uncertainty in regions of poor overlap, **whereas** some other “smoother” Bayesian nonparametric models such as the **Gaussian Process may fare better**.” (Hahn et al. 2020)

BART ≠ GP

- “Given its underlying tree structure, intuitively **BART may not have the flexibility** to capture the additional uncertainty in regions of poor overlap, **whereas** some other “smoother” Bayesian nonparametric models such as the **Gaussian Process may fare better**.” (Hahn et al. 2020)

- “Similarly, while **Gaussian processes may induce smoothness** in the regression, it could be argued **BART-based models are easier to implement** in practice and work well off-the-shelf with minimal tuning.” (Hahn et al. 2020)

BART ≠ GP

- “Given its underlying tree structure, intuitively **BART may not have the flexibility** to capture the additional uncertainty in regions of poor overlap, **whereas** some other “smoother” Bayesian nonparametric models such as the **Gaussian Process may fare better**.” (Hahn et al. 2020)

- “Similarly, while **Gaussian processes may induce smoothness** in the regression, it could be argued **BART-based models are easier to implement** in practice and work well off-the-shelf with minimal tuning.” (Hahn et al. 2020)

- “Note how the **GP-estimated** expected outcomes tick up or down outside the range of the data **based on a handful of observations at the extremes**, as opposed to **BART** and the linear model which **extrapolate in predictable ways**.” (Hahn et al. 2020)

BART ≠ GP

- “Finally, several of the discussants proposed Gaussian process models with **limited discussion of the covariance function and how its parameters are set or inferred**. The covariance function is often pivotal to their success. Unsurprisingly, the squared exponential covariance function performs splendidly on very smooth response surfaces, but what happens when this strong assumption is violated? **By contrast, BART has a long track record of adapting successfully to a wide variety of unknown covariance structures and this robustness** is why we chose to design BCF around BART priors.” (Hahn et. al 2020)

BART ≠ GP

- “Although not widely appreciated, **BART actually is a Gaussian process**, conditional on the trees (integrating over Gaussian priors over the leaf parameters). Specifically, the trees define a covariance function where the correlation between points x and x′ are a function of the proportion of trees in the forest in which the two points occupy the same leaf. **As the number of trees is increased, this covariance function becomes increasingly smooth, although it is singular and nonstationary for a finite number of trees.**” (Hahn et al. 2020)

- (N.B. there are technical errors here)

BART $\longrightarrow$ GP

- $y = \sum_{j=1}^{m} g(x; T_j, M_j) + \varepsilon(x)$

BART $\longrightarrow$ GP

- $y = \sum_{j=1}^{m} g(x; T_j, M_j) + \varepsilon(x)$

- $g(x; T_j, M_j)$ are i.i.d.

BART $\longrightarrow$ GP

- $y = \sum_{j=1}^{m} g(x; T_j, M_j) + \varepsilon(x)$

- $g(x; T_j, M_j)$ are i.i.d.

- These are the hypotheses of the multivariate CLT

BART $\longrightarrow$ GP

- $$y = \sum_{j=1}^{m} g(x; T_j, M_j) + \varepsilon(x)$$
- $g(x; T_j, M_j)$ are i.i.d.
- These are the hypotheses of the multivariate CLT
- As $m \to \infty$: $$\begin{pmatrix} y(x_1) \\ \vdots \\ y(x_n) \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} k_{\text{BART}}(x_1, x_1) & \cdots & k_{\text{BART}}(x_1, x_n) \\ \vdots & \ddots & \vdots \\ k_{\text{BART}}(x_n, x_1) & \cdots & k_{\text{BART}}(x_n, x_n) \end{pmatrix} \right)$$

What is k_{BART}?

- Linero 2017:

- “Proposition 1. Consider the BART model [... under some approximations ...] Then the associated kernel function [...] is given by $k(x, x') \propto \exp(-\lambda \|x - x'\|_1)$.”

What is k_{BART}?

- Linero 2017:

- “Proposition 1. Consider the BART model [... under some approximations ...] Then the associated kernel function [...] is given by $k(x, x') \propto \exp(-\lambda \|x - x'\|_1)$.”

- This is a bog-standard GP covariance function

What is k_{BART}?

- Linero 2017:
- “Proposition 1. Consider the BART model [... under some approximations ...] Then the associated kernel function [...] is given by $k(x, x') \propto \exp(-\lambda \|x - x'\|_1)$.”
- This is a bog-standard GP covariance function
- But: “Furthermore, our experience is that the empirical performance of a minimally-tuned implementation of **BART is frequently better than Gaussian process regression using the equivalent kernel** [...] We conjecture that the reason for BART outperforming Gaussian process regression is that limiting the number of trees in the ensemble allows one to learn a data-adaptive notion of distance between points.”

GAME

• • •

END

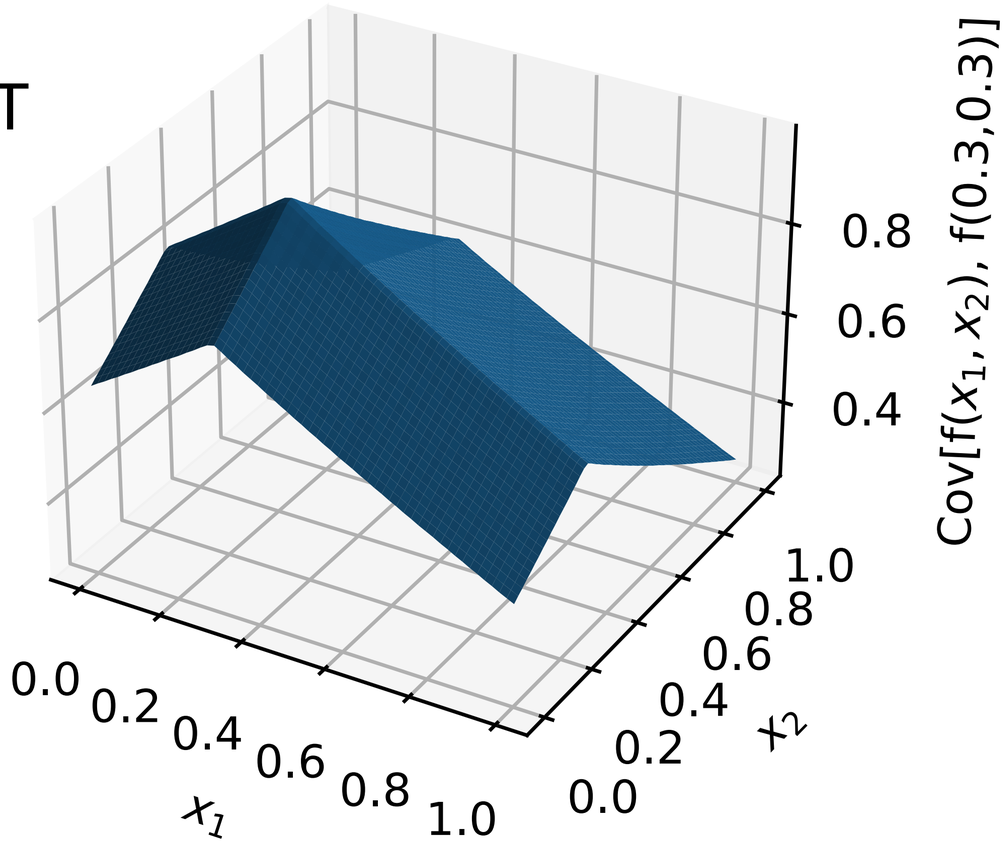
k_{BART} vs. $e^{-\lambda \|x - x'\|_1 / p}$

- Are they similar enough?

k_{BART} vs. $e^{-\lambda \|x - x'\|_1 / p}$

- Are they similar enough?

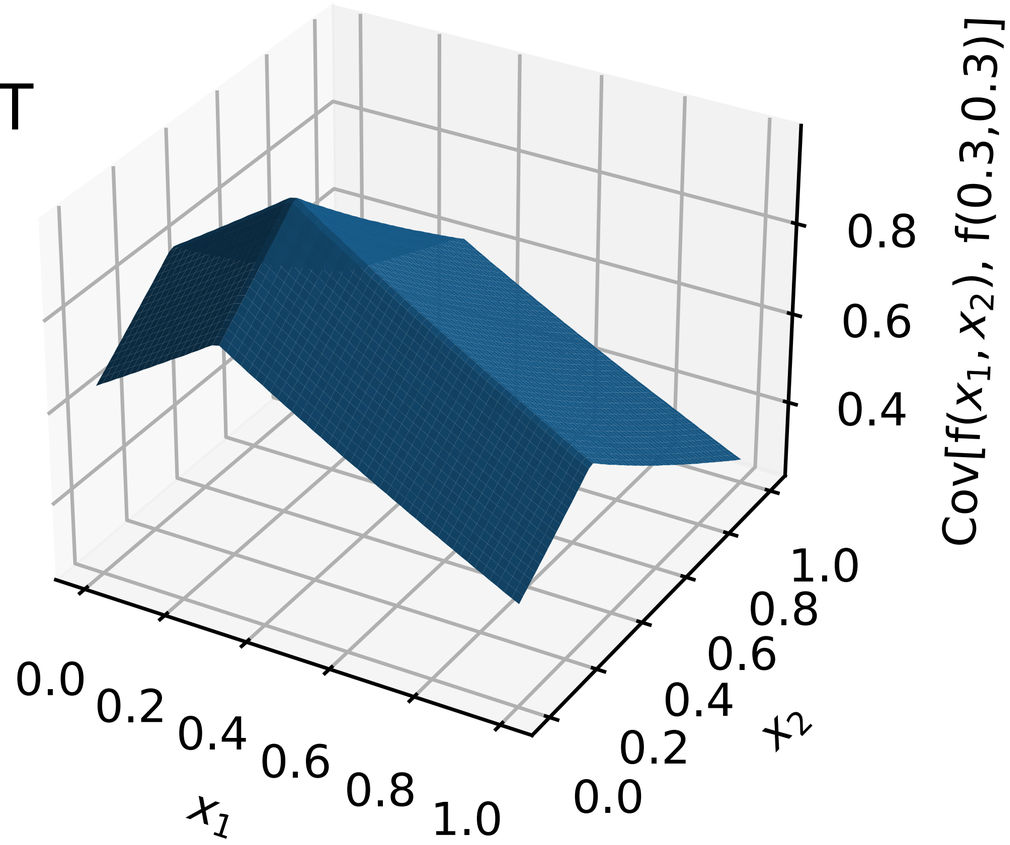
BART



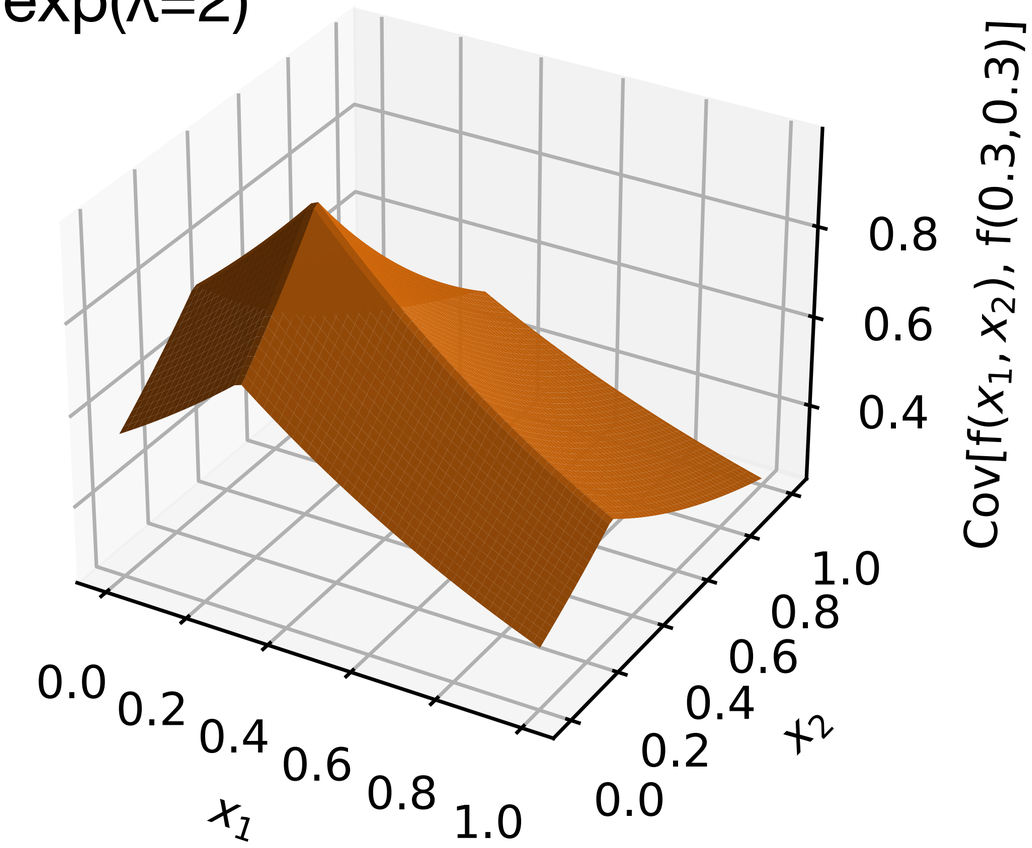
k_{BART} vs. $e^{-\lambda \|x - x'\|_1 / p}$

- Are they similar enough?

BART



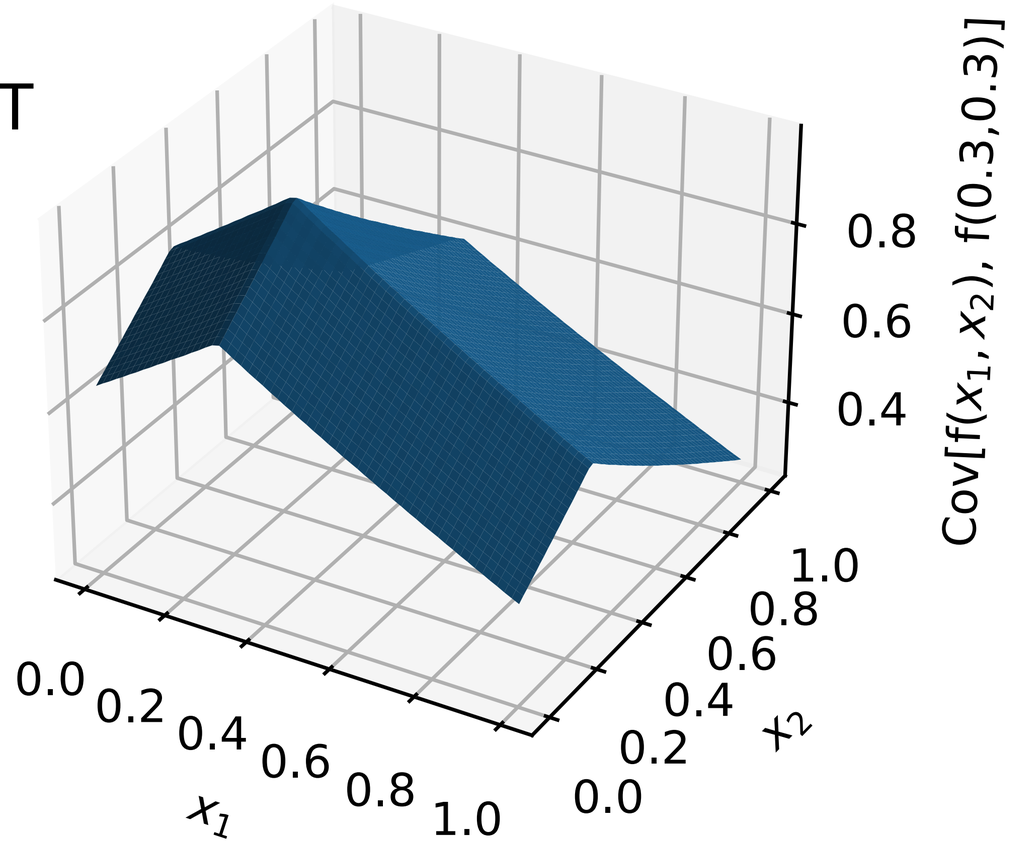
exp(λ=2)



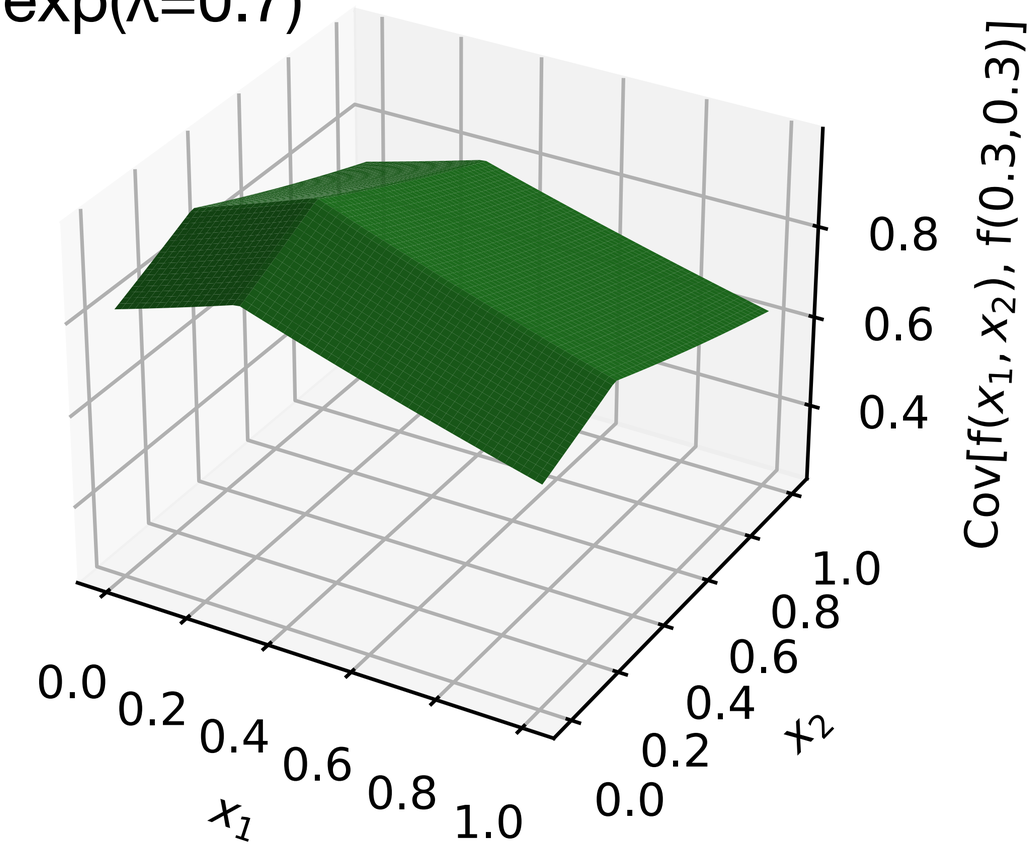
k_{BART} vs. $e^{-\lambda\|x-x'\|_1/p}$

- Are they similar enough?

BART



exp(λ=0.7)



k_{BART} vs. $e^{-\lambda \|x - x'\|_1 / p}$

- Problem:

k_{BART} vs. $e^{-\lambda \|x - x'\|_1 / p}$

- Problem:

- $k_{\text{BART}}(\mathbf{x}, \mathbf{x}') \approx 1 - \dfrac{\|\mathbf{x} - \mathbf{x}'\|_1}{p}$

k_{BART} vs. $e^{-\lambda \|x - x'\|_1 / p}$

- Problem:
- $k_{\text{BART}}(\mathbf{x}, \mathbf{x}') \approx 1 - \dfrac{\|\mathbf{x} - \mathbf{x}'\|_1}{p}$
- $e^{-\lambda \|\mathbf{x} - \mathbf{x}'\|_1 / p} \approx 1 - \lambda \, |x_1 - x_1'| - \lambda \, |x_2 - x_2'|$ if $\lambda \to 0$

k_{BART} vs. $e^{-\lambda \|x - x'\|_1 / p}$

- Problem:
- $k_{\text{BART}}(\mathbf{x}, \mathbf{x}') \approx 1 - \dfrac{\|\mathbf{x} - \mathbf{x}'\|_1}{p}$
- $e^{-\lambda \|\mathbf{x} - \mathbf{x}'\|_1 / p} \approx 1 - \lambda \, |x_1 - x_1'| - \lambda \, |x_2 - x_2'|$ if $\lambda \to 0$
- Either it's not separable, or the intercept prior variance is large

k_{BART} vs. $e^{-\lambda \|x - x'\|_1 / p}$

- Problem:
- $k_{\text{BART}}(\mathbf{x}, \mathbf{x}') \approx 1 - \dfrac{\|\mathbf{x} - \mathbf{x}'\|_1}{p}$
- $e^{-\lambda \|\mathbf{x} - \mathbf{x}'\|_1 / p} \approx 1 - \lambda \, |x_1 - x_1'| - \lambda \, |x_2 - x_2'| \qquad \text{if} \qquad \lambda \to 0$
- Either it's not separable, or the intercept prior variance is large
- Speculative solution: $\exp(-\lambda \|x - x'\|_1 / p) - e^{-\lambda}$, which is p.s.d. although not known

$$\exp(-\lambda \|x - x'\|_1 / p) - e^{-\lambda}$$

- Proof of positivity:

$\exp(-\lambda \|x - x'\|_1 / p) - e^{-\lambda}$

- Proof of positivity:
- $e^{\lambda k} = \sum_{n=0}^{\infty} \frac{(\lambda k)^n}{n!}$

$\exp(-\lambda \|x - x'\|_1 / p) - e^{-\lambda}$

- Proof of positivity:
- $e^{\lambda k} = \sum_{n=0}^{\infty} \frac{(\lambda k)^n}{n!}$ $\qquad$ $e^{k} - 1 = \sum_{n=1}^{\infty} \frac{(\lambda k)^n}{n!}$
- $\exp(\lambda k(x, x'))$ is a valid covariance function

$\exp(-\lambda \|x - x'\|_1 / p) - e^{-\lambda}$

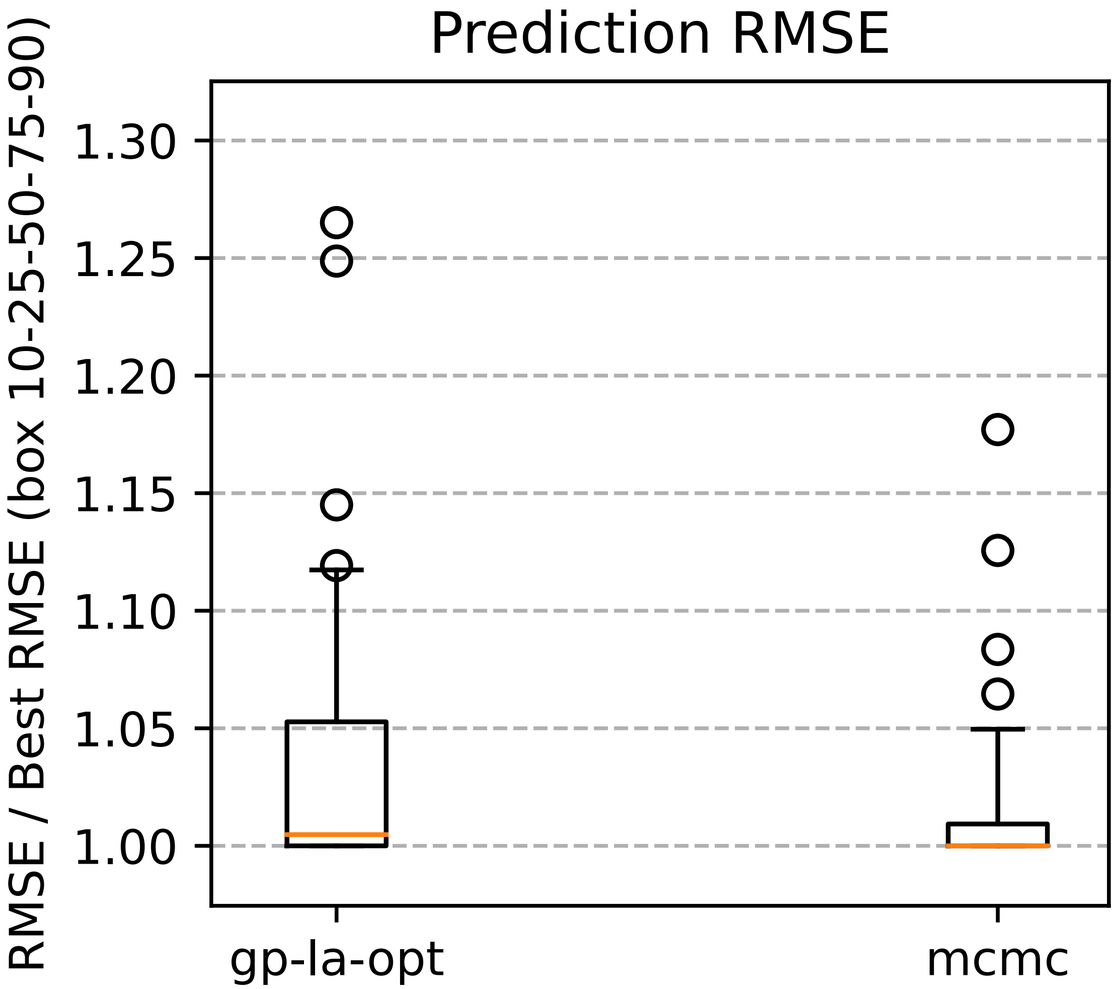
- Proof of positivity:
- $e^{\lambda k} = \sum_{n=0}^{\infty} \frac{(\lambda k)^n}{n!}$ $\qquad$ $e^{k} - 1 = \sum_{n=1}^{\infty} \frac{(\lambda k)^n}{n!}$
- $\exp(\lambda k(x, x'))$ is a valid covariance function
- plug $k(x, x') = \frac{1}{p} \sum_{i=1}^{p} (1 - |x_i - x_i'|)$

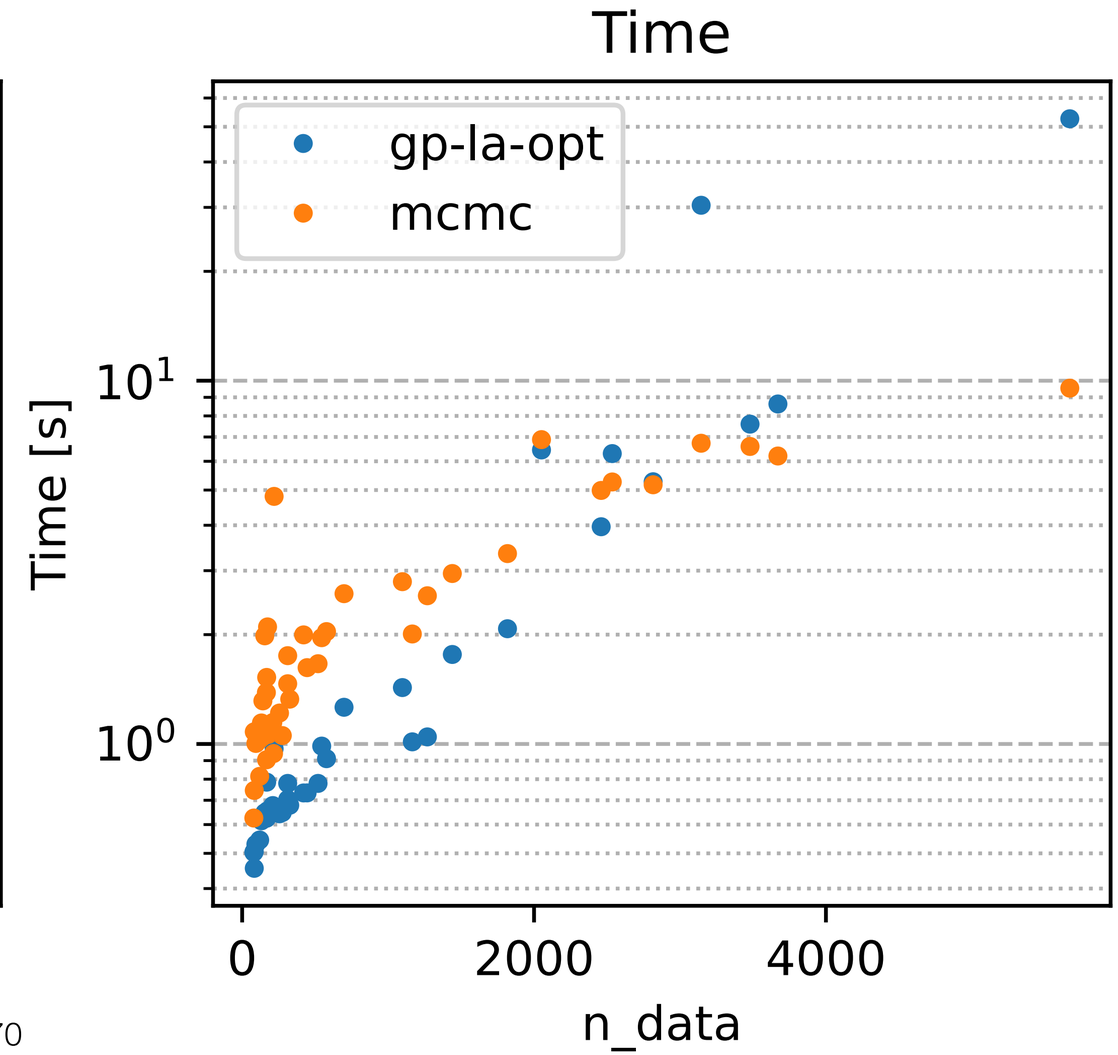
$\exp(-\lambda \|x - x'\|_1 / p) - e^{-\lambda}$

- Proof of positivity:
- $e^{\lambda k} = \sum_{n=0}^{\infty} \frac{(\lambda k)^n}{n!}$ $\qquad$ $e^{k} - 1 = \sum_{n=1}^{\infty} \frac{(\lambda k)^n}{n!}$
- $\exp(\lambda k(x, x'))$ is a valid covariance function
- plug $k(x, x') = \frac{1}{p} \sum_{i=1}^{p} (1 - |x_i - x_i'|)$ (triangular covariance function)

BART vs. GP

Fixed hyperparameters only, WIP hyperparameter tuning and other comparisons





My k_{BART}

$$k(\mathbf{x}, \mathbf{x}') = k_0(\mathbf{n}^-, \mathbf{n}^0, \mathbf{n}^+), \qquad \mathbf{n} = \mathbf{n}^- + \mathbf{n}^0 + \mathbf{n}^+,$$
$$k_d(\mathbf{0}, \mathbf{0}, \mathbf{0}) = k_d((), (), ()) = 1,$$

$$k_d(\mathbf{n}^-, \mathbf{n}^0, \mathbf{n}^+) = 1 - P_d \left[1 - \frac{1}{W(\mathbf{n})} \sum_{\substack{i=1 \\ n_i \neq 0}}^{p} \frac{w_i}{n_i} \left(\sum_{k=0}^{n_i^- - 1} k_{d+1}(\mathbf{n}^-_{n_i^- = k}, \mathbf{n}^0, \mathbf{n}^+) + \sum_{k=0}^{n_i^+ - 1} k_{d+1}(\mathbf{n}^-, \mathbf{n}^0, \mathbf{n}^+_{n_i^+ = k}) \right) \right],$$

$$W(\mathbf{n}) = \sum_{\substack{i=1 \\ n_i \neq 0}}^{p} w_i, \qquad \mathbf{w} > 0, \qquad P_d = \frac{\alpha}{(1+d)^\beta},$$

My k_{BART}

$$k^D_{D-2}(\mathbf{n}^-, \mathbf{0}, \mathbf{n}^+) = 1,$$

$$k^D_{D-2}(\mathbf{n}^-, \underbrace{\mathbf{n}^0}_{\neq \mathbf{0}}, \mathbf{n}^+) = 1 - P_{D-2}\left[1 - \frac{1}{W(\mathbf{n})}\left[(1 - P_{D-1})S + P_{D-1}(1 - (1-\gamma)P_D)\sum_{\substack{i=1 \\ n_i \neq 0}}^{p} \frac{w_i}{n_i}\Bigg(\right.\right.$$

$$\left\{ n_i^- \,\middle|\, \frac{1}{W(\mathbf{n}_{n_i^-=0})}\left(S - w_i\frac{n_i^- + n_i^+}{n_i} + w_i\left\{ n_i^0 + n_i^+ \,\middle|\, \frac{n_i^+}{n_i^0 + n_i^+}\right\}\right) + \right.$$

$$\left. + \frac{n_i^- - 1}{W(\mathbf{n})}\left(S + w_i\frac{n_i^0}{n_i}\right) - \frac{w_i n_i^0}{W(\mathbf{n})}(\psi(n_i) - \psi(1 + n_i^0 + n_i^+))\right\} +$$

$$+ \left\{ n_i^+ \,\middle|\, \frac{1}{W(\mathbf{n}_{n_i^+=0})}\left(S - w_i\frac{n_i^+ + n_i^-}{n_i} + w_i\left\{ n_i^0 + n_i^- \,\middle|\, \frac{n_i^-}{n_i^0 + n_i^-}\right\}\right) + \right.$$

$$\left.\left.\left. + \frac{n_i^+ - 1}{W(\mathbf{n})}\left(S + w_i\frac{n_i^0}{n_i}\right) - \frac{w_i n_i^0}{W(\mathbf{n})}(\psi(n_i) - \psi(1 + n_i^0 + n_i^-))\right\}\right)\right]\right],$$

$$S = \sum_{\substack{i=1 \\ n_i \neq 0}}^{p} w_i\left(1 - \frac{n_i^0}{n_i}\right),$$

$$\{x \mid E\} = \begin{cases} E & x > 0, \\ 0 & x = 0, \text{ even if } E \text{ is not well defined,} \end{cases}$$

Code

- My GP python package: https://github.com/Gattocrucco/lsqfitgp
- Implements the BART kernel