Bayesian Regression Tree Models!!!

M. T. Pratola
Dept. of Statistics
The Ohio State University
Email: mpratola@stat.osu.edu
Web: www.matthewpratola.com

February 6, 2014

A Long Time Ago In A Galaxy Far Far Away...

A Long Time Ago In A Galaxy Far Far Away....

Classification And Regression Trees!

by: Leo S. Breiman

1928–2005

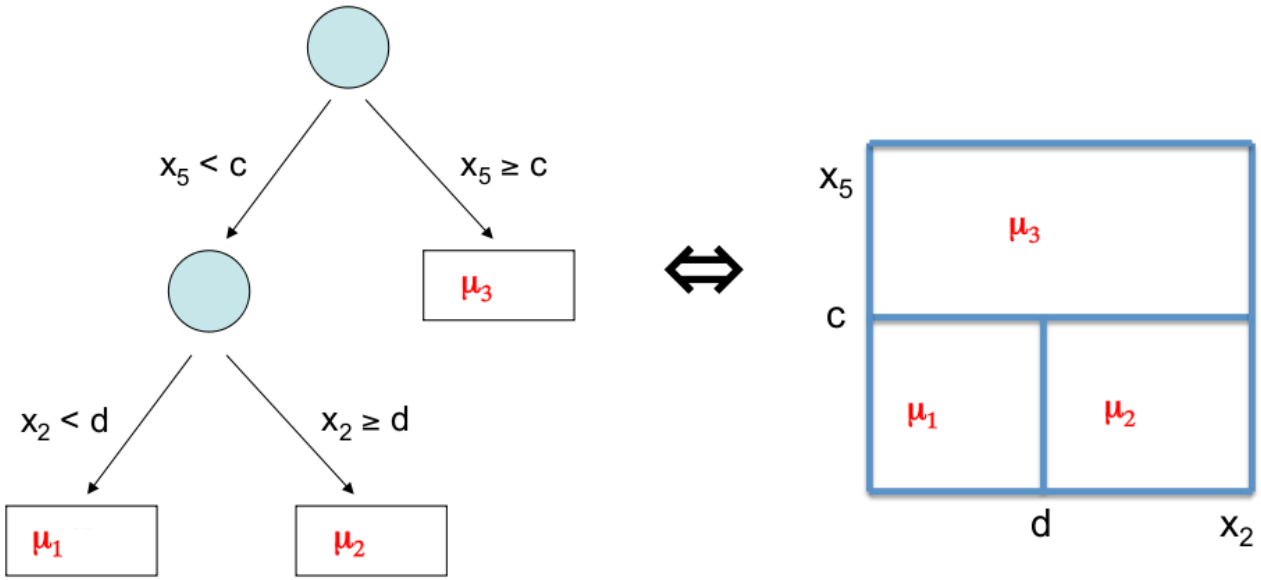
(and others...)

CART: Classification And Regression Trees

★ Leo Breiman, Jerome H. Friedman, Richard A. Olshen and Charles J. Stone, "Classification and Regression Trees", Wadsworth International Group, 1984.

- Problem: Simple regression models often don't work well with complicated real-world data
- Idea: Fit simple regression models to different regions of covariate space to get a good overall fit to the data
- Solution: Partition the covariate space to fit the different models using a binary classification tree.
- How: Partitions try to increase fit to data subject to a complexity constraint. Bayesian in flavor, but in a fairly ad-hoc manner

CART: Classification And Regression Trees



Here, each $\mu_i \equiv \mu_i(x)$ can be a unique regression function.
But, we only get 1 tree...

Bayesian CART

★ Hugh A. Chipman, Edward I. George, and Robert E. McCulloch, "Bayesian cart model search", Journal of the American Statistical Association, vol.93, pp.935-960, 1998.
★ Hugh A. Chipman, Edward I. George, and Robert E. McCulloch, "Hierarchical priors for bayesian cart shrinkage", Statistics and Computing, vol.10, pp.17-24, 2000.
★ Hugh A. Chipman, Edward I. George, and Robert E. McCulloch, "Bayesian treed models", Machine Learning, vol.48, pp.299-320, 2002.
★ David G. Denison, Bani K. Mallick, and Adrian F. M. Smith, "A bayesian cart algorithm", Biometrika, vol.85, pp.363-377, June 1998.

- Place CART within a Bayesian framework by specifying a prior on tree space.
- Can get multiple tree realizations by using tree-changing proposal distribution: birth/death/change/swap.
- Get multiple realizations of 1 tree, average over posterior to form predictions.

Bayesian Regression Tree Models

Data generating model is

$$y(x) = f(x) + \epsilon, \quad \epsilon \sim N(0, \sigma^2)$$

A regression tree models this data as

$$y(x) = g(x; T, M) + \epsilon$$

where $g(\cdot; T, M)$ represents the regression tree

Bayesian Regression Tree Models

Data generating model is $y(x) = f(x) + \epsilon, \quad \epsilon \sim N(0, \sigma^2)$

A regression tree models this data as

$$y(x) = g(x; T, M) + \epsilon$$

where $g(\cdot; T, M)$ represents the regression tree.

Bayesian framework:

$$\pi(T, M, \sigma^2) = \pi(M|T, \sigma^2)\pi(T|\sigma^2)\pi(\sigma^2)$$

see Chipman et al (1998), Denison et al (1998)

From Bayesian CART to BART

★ Yuhong Wu, Hkon Tjelmelanda and Mike West, "Bayesian CART: Prior Specification and Posterior Simulation", Journal of Computational and Graphical Statistics, vol.16, 2007.
★ Matt Taddy, Robert B. Gramacy and Nick Polson, "Dynamic Trees for Learning and Design", Journal of the American Statistical Association, vol.106, pp.109-123, 2010.
★ Robert B. Gramacy and Herbert K.H. Lee, "Bayesian Treed Gaussian Process Models With an Application to Computer Modeling", Journal of the American Statistical Association, vol.103, pp.1119-1130, 2008.
★ Hugh A. Chipman, Edward I. George and Robert E. McCulloch, "BART: Bayesian Additive Regression Trees", The Annals of Applied Statistics, vol.4, pp.266-298, 2010.

Bayesian Additive Regression Tree Models

BART model is similar:
A regression tree models this data as

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

where $g_j(\cdot; T_j, M_j)$ represents a single regression tree.

Bayesian framework:

$$\pi((T_1, M_1), \ldots, (T_m, M_m), \sigma^2) \prod_{j=1}^{m} \pi(M_j | T_j, \sigma^2) \pi(T_j | \sigma^2) \pi(\sigma^2)$$

see Chipman et al (2010)

Regression Trees

- $g(x; T, M)$ is a regression tree f'n that assigns the map $\mu(x)$ to a given input x

Regression Trees

- $g(x; T, M)$ is a regression tree f'n that assigns the map $\mu(x)$ to a given input x
- Tree is parameterized by
  - T denotes the tree structure (decision rules, depth)
  - $M = (\mu_1, \ldots, \mu_b)$ denotes the bottom-node μ's

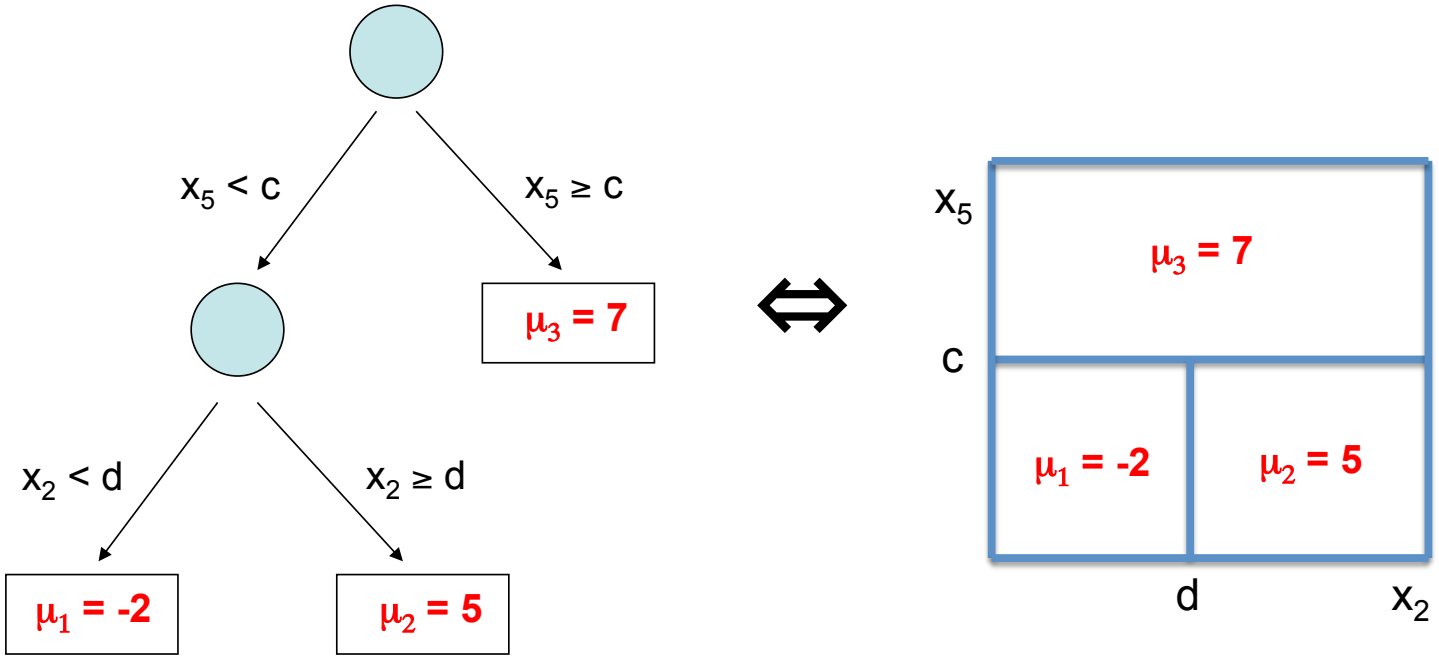
Regression Trees

- $g(x; T, M)$ is a regression tree f'n that assigns the map $\mu(x)$ to a given input x
- Tree is parameterized by
  - T denotes the tree structure (decision rules, depth)
  - $M = (\mu_1, \ldots, \mu_b)$ denotes the bottom-node μ's
- Many forms for $\mu_i(x) \in M$
  - linear: $\mu(x) = x'\beta$ (Chipman et al 1998; Denison et al 1998)
  - Gaussian Process: $\mu(x) \sim GP(x; \cdot)$ (Gramacy and Lee, 2008)
  - Constant: $\mu(x) = \mu$ (Wu et al 2007; Chipman et al 2010)

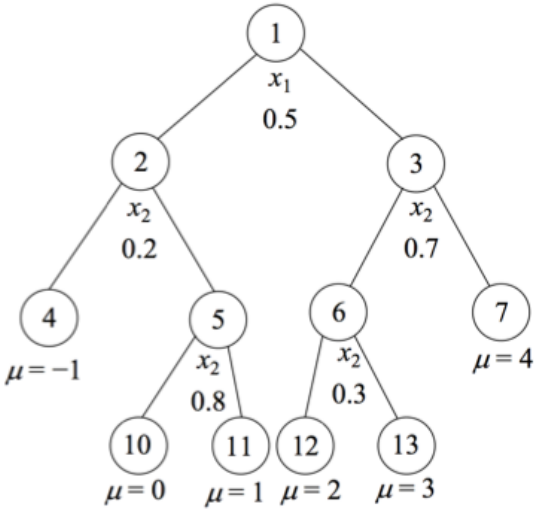
Regression Trees

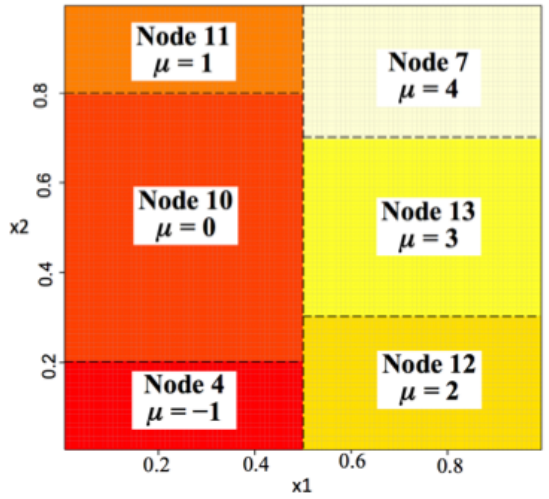
- $g(x; T, M)$ is a regression tree f'n that assigns the map $\mu(x)$ to a given input x
- Tree is parameterized by
  - T denotes the tree structure (decision rules, depth)
  - $M = (\mu_1, \ldots, \mu_b)$ denotes the bottom-node μ's
- Many forms for $\mu_i(x) \in M$
  - linear: $\mu(x) = x'\beta$ (Chipman et al 1998; Denison et al 1998)
  - Gaussian Process: $\mu(x) \sim GP(x; \cdot)$ (Gramacy and Lee, 2008)
  - Constant: $\mu(x) = \mu$ (Wu et al 2007; Chipman et al 2010)
- Typically considering conjugate forms so that $\pi(T|\sigma^2) = \int \pi(T|M, \sigma^2)\pi(M)dM$ is available in closed form

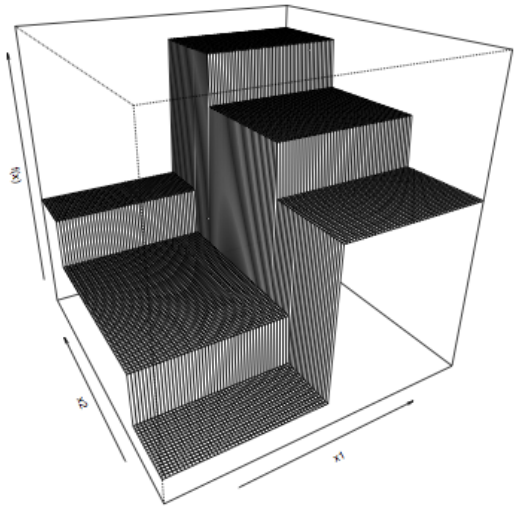
Regression Trees



Regression Trees







Three different views of a bivariate single tree.

source: E.I. George, BNPSki2014

Building up fit by adding tiny bits of fit...

pointilism=Seurat, modern pointilism=ANSI art?

MCMC Algorithm

Draw $T, M|\cdot$ in two steps:

1. draw $T|\cdot$ (Metropolis-Hastings step via proposal distributions)
2. draw $M|T, \cdot$ (Gibbs step for conjugate priors)

Draw $\sigma|M, T, \cdot$ (Gibbs step for conjugate prior)

The good, the bad

The Good:

- Flexible model as the "basis" adapts to the data. Handling continuous and discrete variables is straightforward
- Scales to large datasets using a parallel MCMC sampler (Pratola et al.)

The Bad:

- Bayesian regression tree models known to suffer from poor mixing due to the MH step for $T|\cdot$
- Leads to lack of interpretability of regression trees, under-representation of uncertainty, and more complicated problems in more complicated models

Bayesian Regression Trees in Computer Experiments

★ Robert B. Gramacy, Matt Taddy, and Stefan M. Wild, "Variable selection and sensitivity analysis using dynamic trees, with an application to computer code performance tuning", The Annals of Applied Statistics, vol.7, 2013.
★ Hugh A. Chipman, Pritam Ranjan and Weiwei Wang, "Sequential design for computer experiments with a flexible Bayesian additive model", The Canadian Journal of Statistics, vol.40, pp.663?678, 2012.
★ Matthew T. Pratola, Hugh A. Chipman, James Gattiker, David M. Higdon, Robert McCulloch and William Rust, "Parallel Bayesian Additive Regression Trees", Journal of Computational and Graphical Statistics, to appear.
★ Matthew T. Pratola and David M. Higdon, "Bayesian Regression Tree Calibration of Complex High-Dimensional Computer Models", revised

And not in computer experiments, but maybe still useful...

★ Matthew T. Pratola, "Efficient Metropolis-Hastings Proposal Mechanisms for Bayesian Regression Tree Models", submitted
★ Edward I. George, Hugh A. Chipman, Robert McCulloch and Tom Shively, "Monotone BART", BNPSki, 2014.
★ Christoforos Anagnostopoulos and Robert B. Gramacy, "Dynamic Trees for Streaming and Massive Data Contexts", tech report, University of Chicago Booth School of Business, 2012.
★ Justin Bleich, Adam Kapelner, Edward I. George and Shane T. Jensen, "Variable Selection Inference for Bayesian Additive Regression Trees", submitted

more of the bad...

Previous attempts to improve mixing:

1. Early literature suggests augmenting birth/death proposals with change and swap proposals, but they are very inefficient.
2. Multiple chain/multiple restart approaches
3. Chipman et al (2010) use an additive representation which forces shallow trees. It was believed that in such a setup, birth/death proposals would be sufficient to ensure adequate mixing.
4. Wu et al (2007) develop a "radical restructure" proposal which seems to alleviate mixing problems in their examples. However, it is computationally expensive and does not scale well with p, the number of predictors.
5. Gramacy and Lee (2008) suggest a SMC approach.

When Mixing Goes Wrong

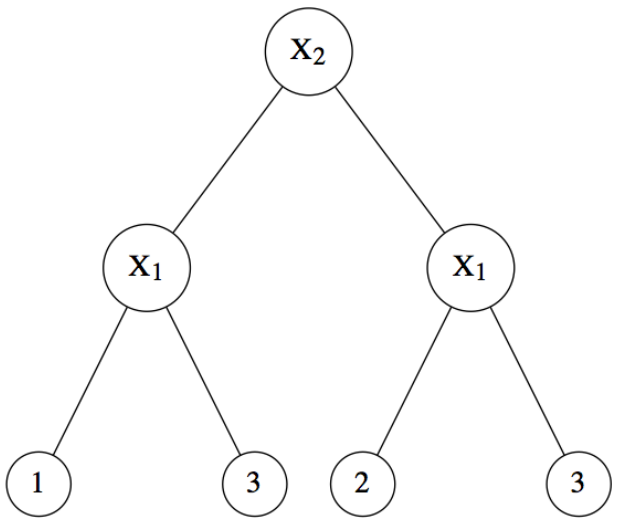
Let's look at three examples

1. A single-tree example given in Wu et al 2007
2. A computer experiments example using BART
3. A calibration example from Pratola & Higdon, 2014

Single-tree example

Wu et al generate data according to the following function, which defines a response surface with 3 regions. In this setup, x_1, x_3 are generated to be highly correlated.

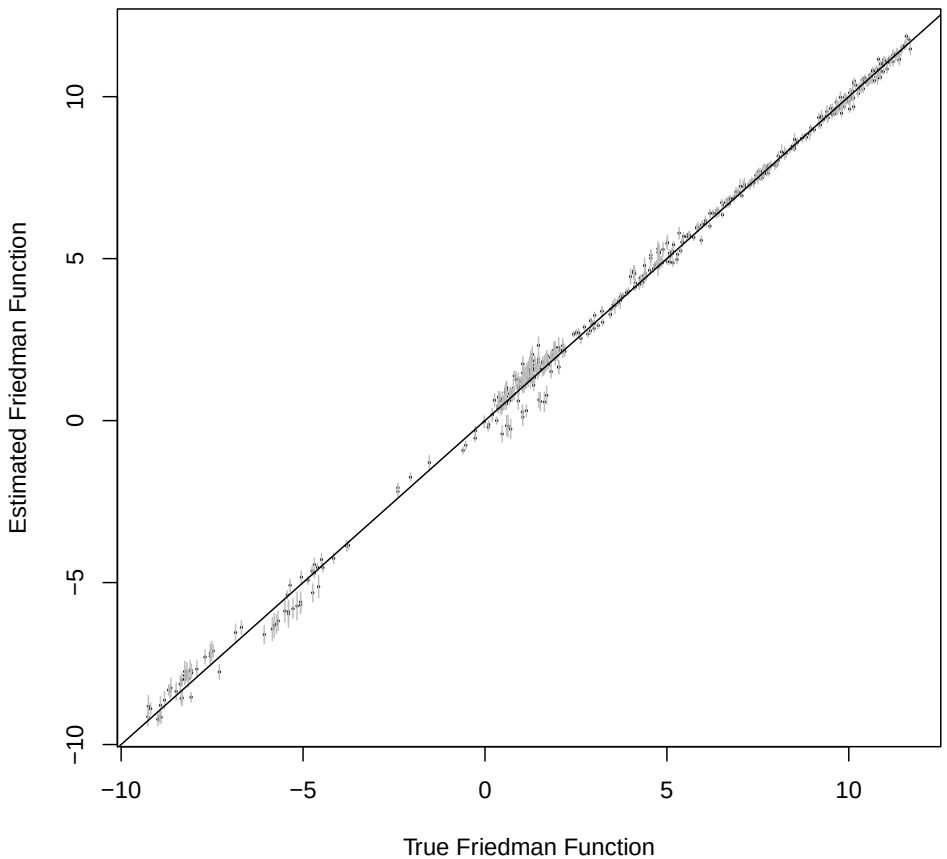
$$y(x) = \begin{cases} 1 + N(0, 0.25) & \text{if } x_1 \leq 0.5 \text{ and } x_2 \leq 0.5 \\ 3 + N(0, 0.25) & \text{if } x_1 \leq 0.5 \text{ and } x_2 > 0.5 \\ 5 + N(0, 0.25) & \text{if } x_1 > 0.5 \end{cases}$$



Computer Experiments Example

BART model is fit to the Friedman function with $n = 5k$ and small $\sigma^2 = 0.1$ to simulate a computer experiments dataset:

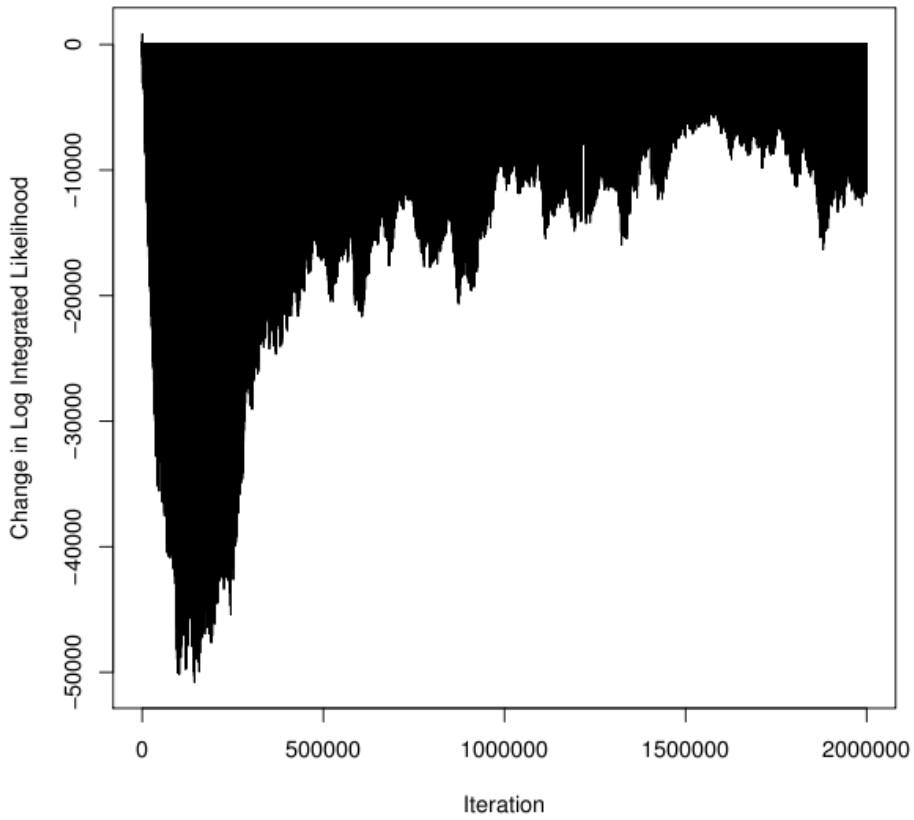
$$f(x) = 10sin(2\pi x_1 x_2) + (x_3 - .5)^2 + x_4 + x_5$$



Computer Experiments Example

BART model is fit to the Friedman function with a small $\sigma^2 = 0.1$ to simulate a computer experiments dataset:

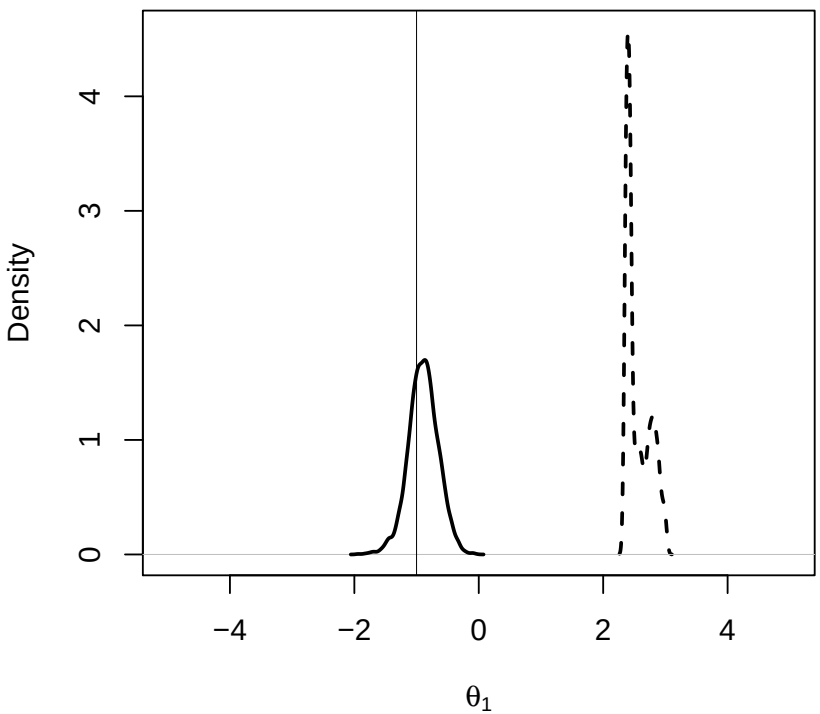
$$f(x) = 10sin(2\pi x_1 x_2) + (x_3 - .5)^2 + x_4 + x_5$$

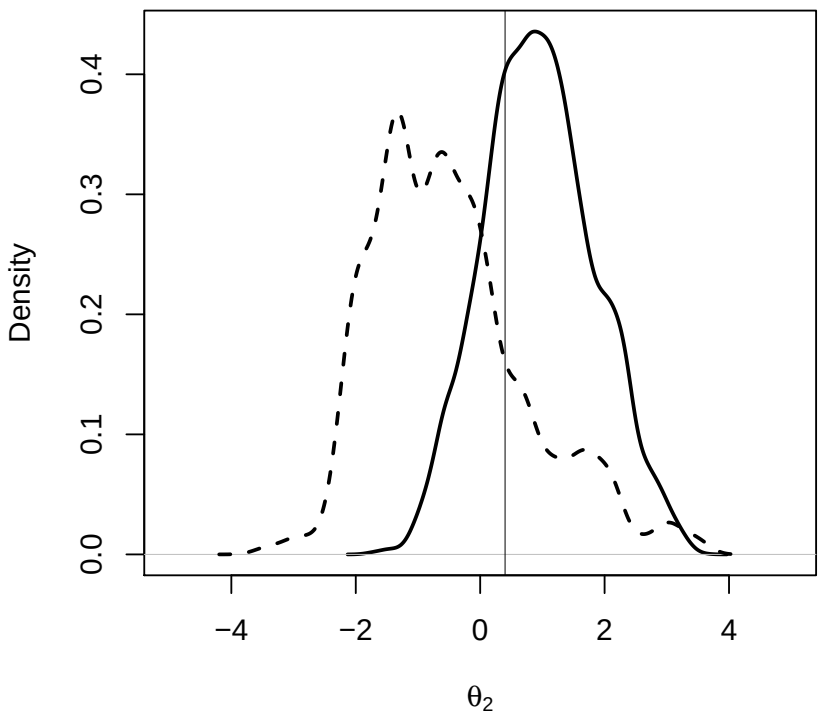


Calibration Example

Pratola and Higdon (2014) develop a CMCE model using additive regression trees to combine field data and simulator outputs for estimating simulator parameters θ and predicting the field at out-of-sample input settings.

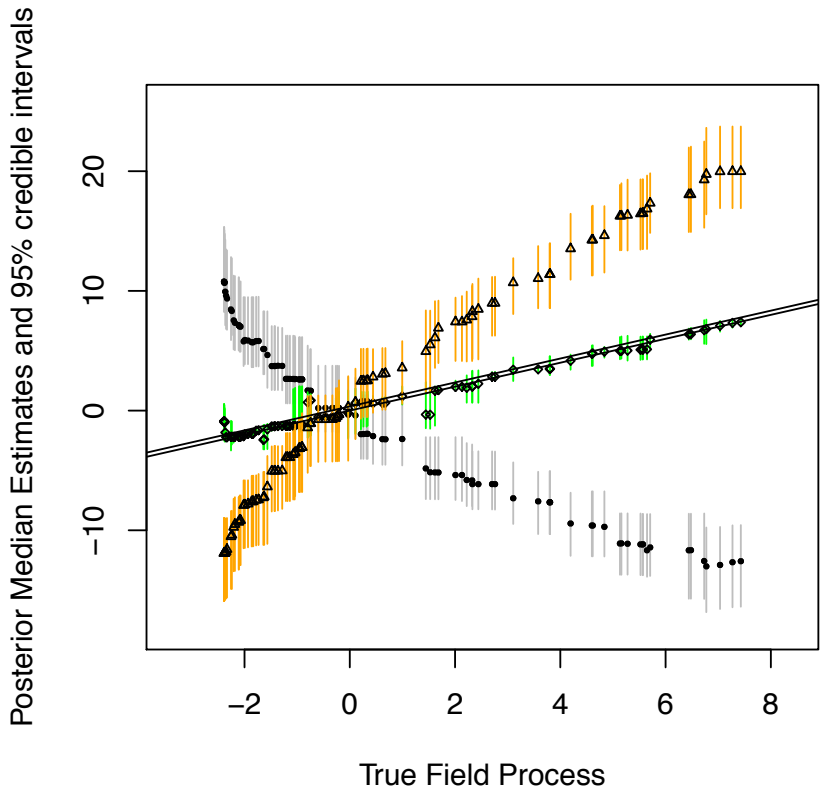
$$y_f(x) = \eta(x, \theta) + \delta(x) + \epsilon_f; \quad y(x, t) = \eta(x, t) + \epsilon$$



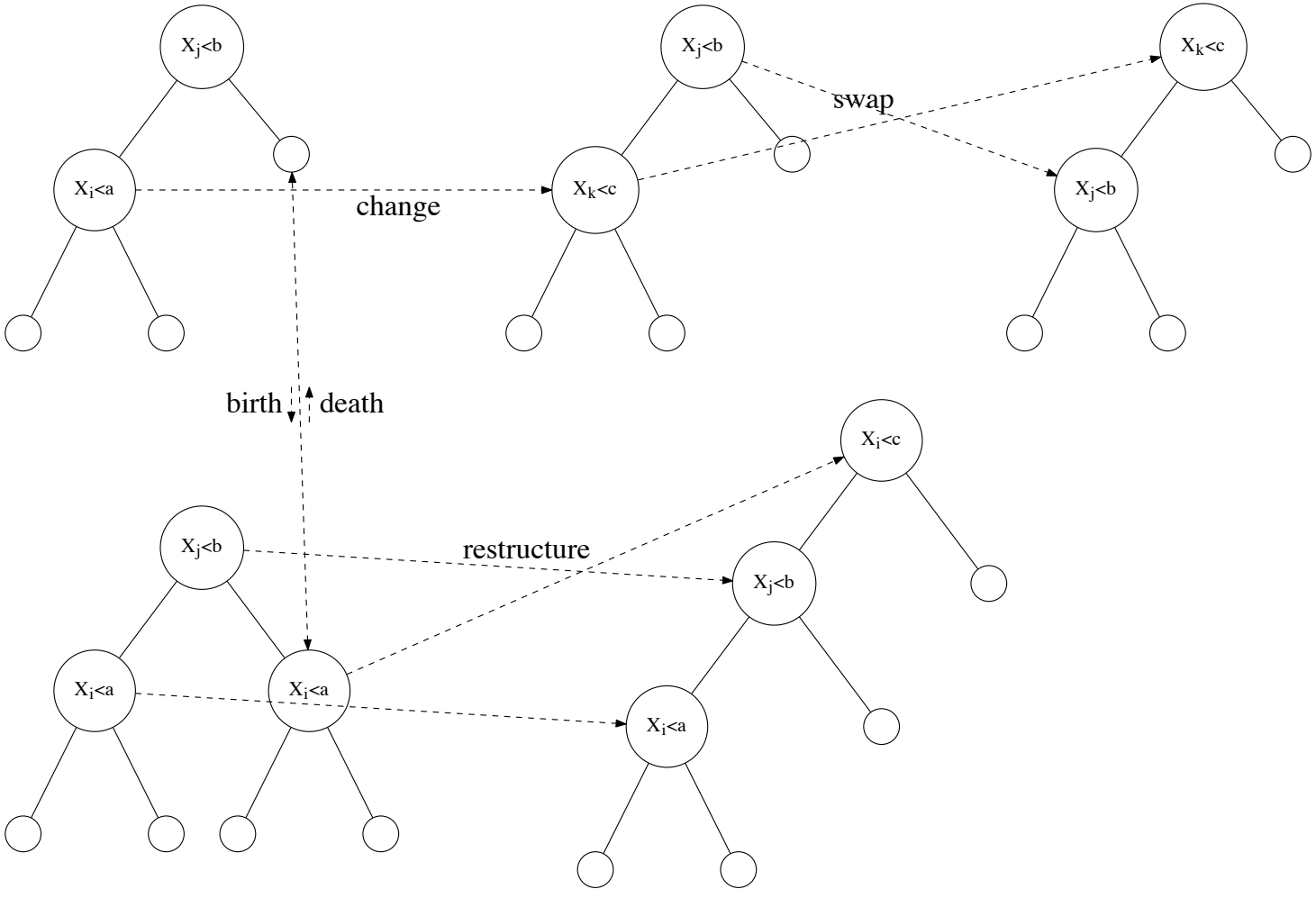


Calibration Example

Here, $\eta(x,t) = t_1 x + t_2$ and $y_f(x) = -x + 0.4 + 0.1x^2 + \epsilon_f$ where $\epsilon_f \sim N(0, 0.01)$



Existing Proposals



New, Efficient Proposal Mechanisms

Structural changes in the tree are important to the MCMC sampler.. most of the "fit" is realized through birth/death changes to trees in BART

But what about the uncertainty? Some of our results suggest that we miss roughly half of the posterior uncertainty when structural proposals are poor.

Our work lead to 2 novel proposal mechansims:

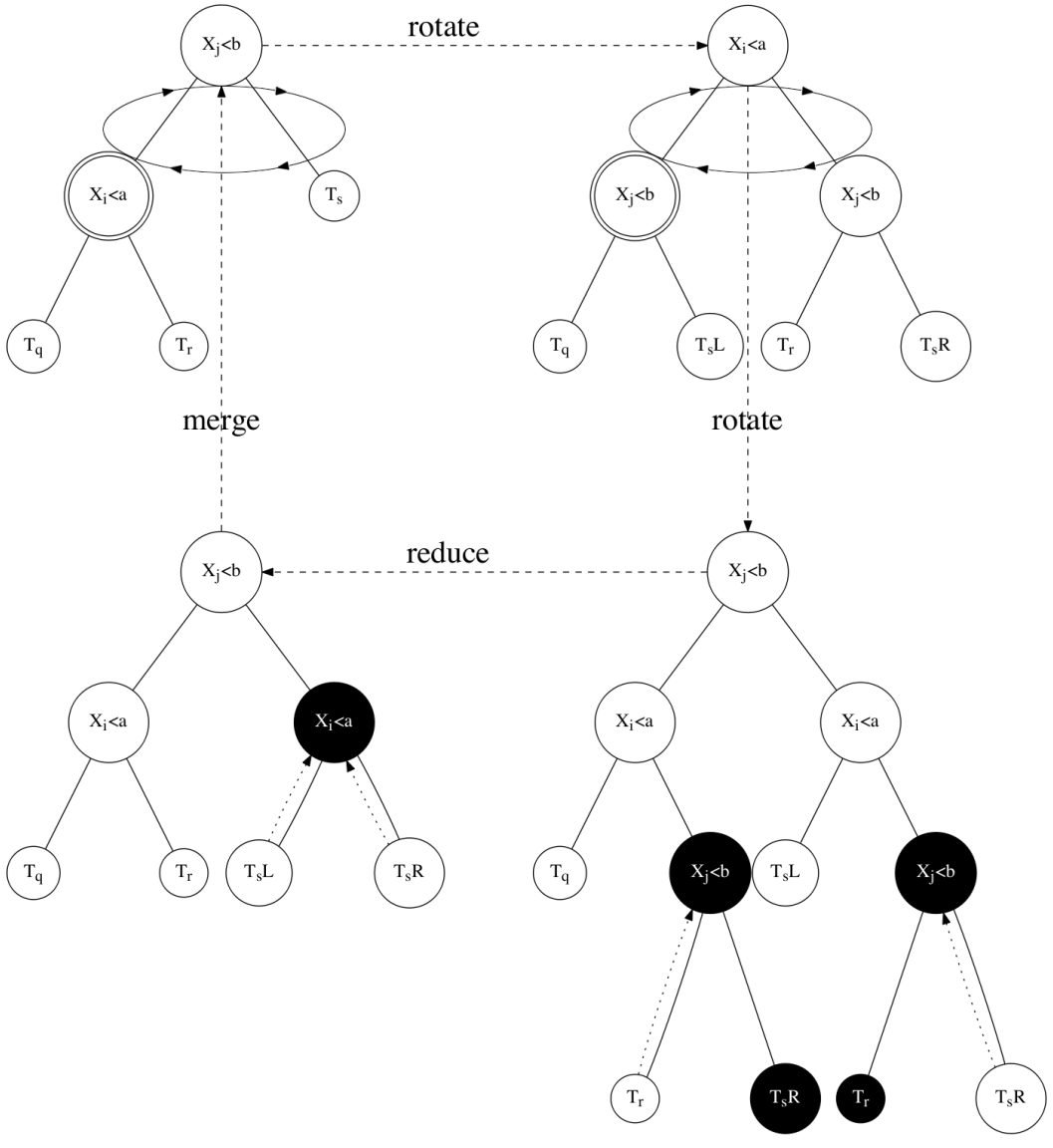
1. Tree rotation
2. Perturb and perturb within change-of-variable

Tree Rotation

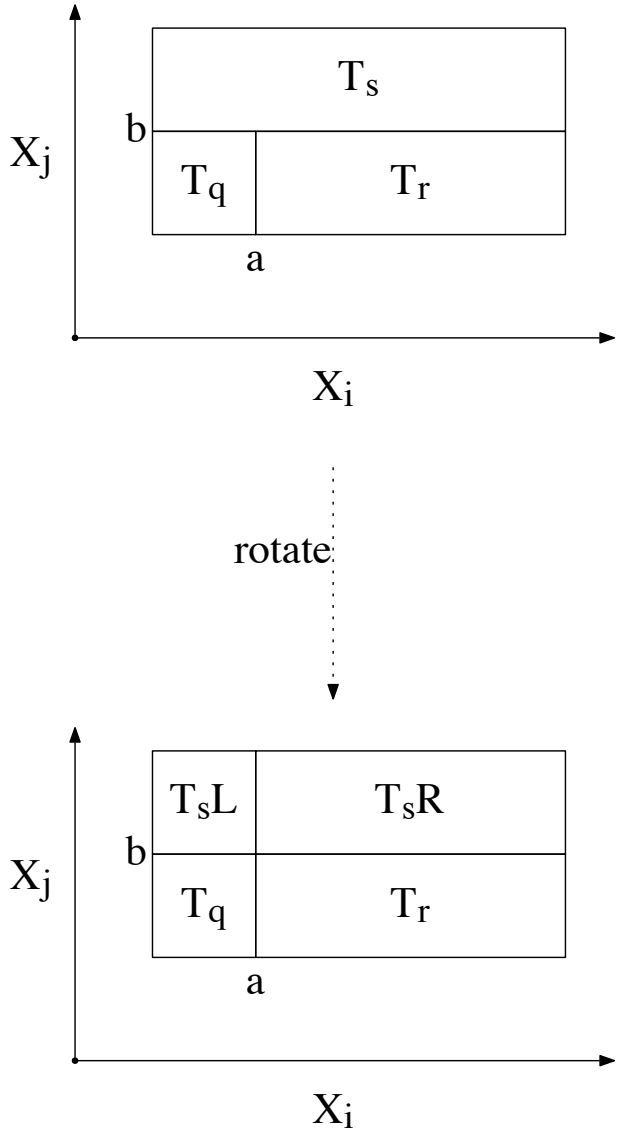
So far, we have devised proposal mechanisms to efficiently modify an existing tree structure

But, what about trees that are structurally different that still have high posterior probability? How can we efficiently generate such trees in our MCMC sampler?

Tree Rotation



Tree Rotation



Tree Rotation

For the rotate step, we

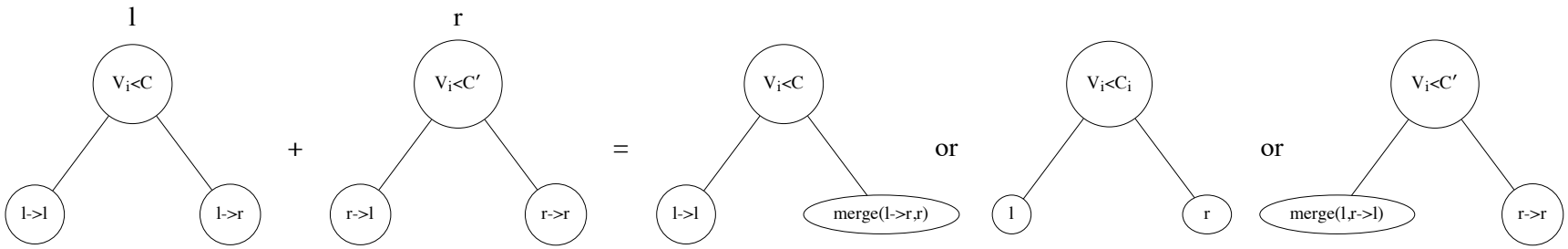
1. Generate $T' \sim q(T, \cdot)$
2. Accept T' with probability $\alpha = min\left\{1, \frac{\pi(T')q(T',T)}{\pi(T)q(T,T')}\right\}$

- This involves the ratio of the integrated likelihoods for all the terminal nodes belonging to the subtree of the rotation nodes parent.
- Rotation is composition of simpler operations. Right-rotation:

$$\mathcal{R}T = \mathcal{R}^L_{merge}\mathcal{R}^R_{merge}\mathcal{R}^L_{cut}\mathcal{R}^R_{cut}\mathcal{R}^R_{rot}T$$

Generating Rotation Proposals

- $\mathcal{R}_{rot}$ can occur at any internal node exluding the root node
- $\mathcal{R}_{cut}$ is deterministic
- $\mathcal{R}_{merge}$ is defined recursively and can lead to the generation of a finite number of merged trees, of which one is randomly selected
  - There are 7 unique merge types in this recursion, for example:

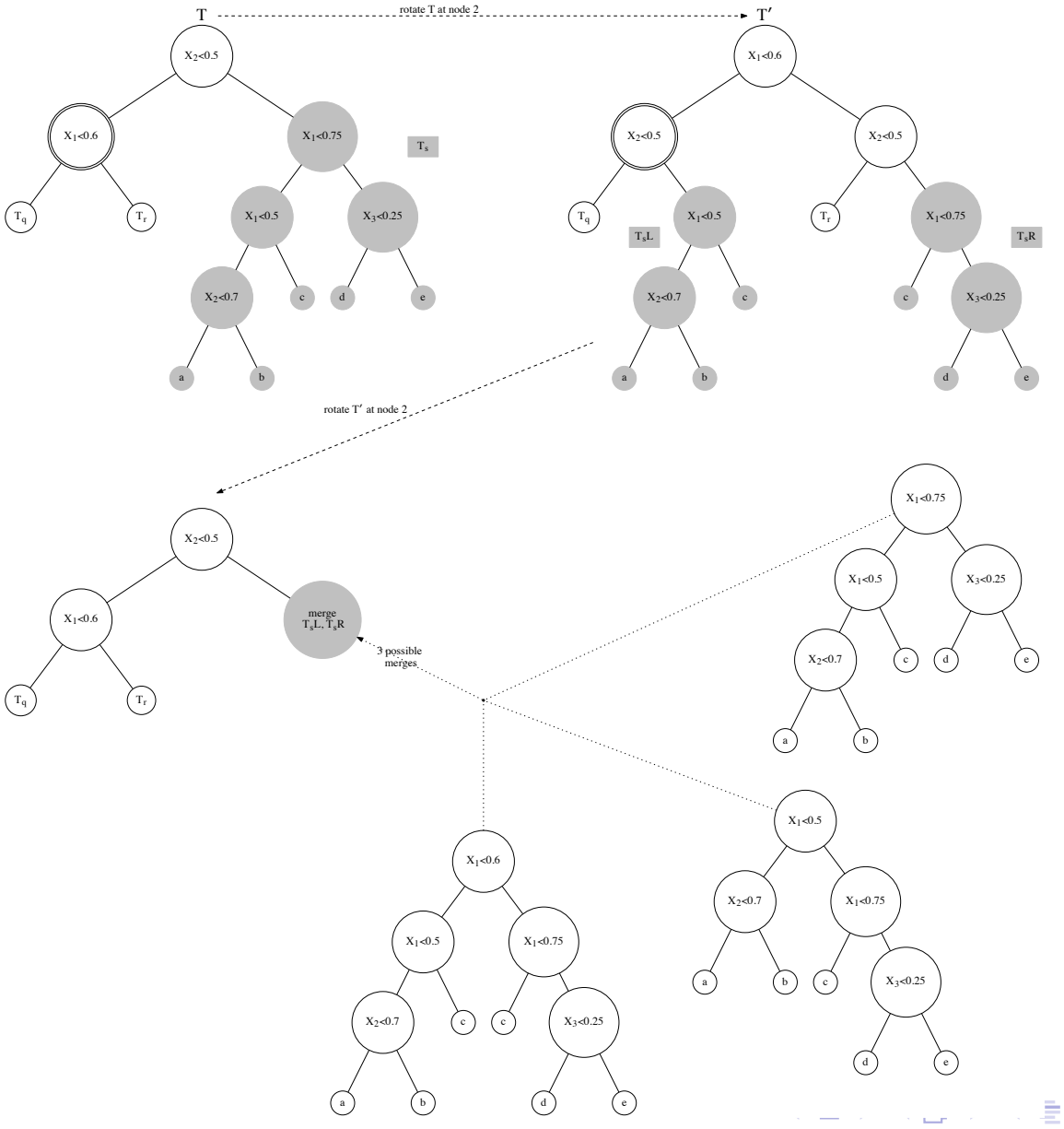


Tree Rotation

Benefits include:

- moves between high probability modes
- 1 rotate restructures the tree in a way that would have taken multiple birth/death moves
- pushes some nodes up and others down increasing the chance that an internal node can get low enough to be pruned
- changes model dimension, no existing MH proposal for internal nodes does this
- it remains a local computation, so cheaper to implement than the restructure move

Rotation Example



Perturb

Previous approaches to propose a new cutpoint value c_i at node i:

- Draw from the prior. Results in low acceptance rates because proposed cutpoint is often not consistent with existing tree structure
- Draw from prior restricted to structure of tree ancestral to node i. Often still has low acceptance rates as only partially consistent with existing tree structure

Perturb

Let $C^{v_i}_{p(i)}$ be the collection of cutpoints for all nodes ancestral i splitting on v_i
Let $C^{v_i}_{l(i)}$ (similarly $C^{v_i}_{r(i)}$) be the collection of cutpoints for all left (similarly right) descendent nodes of i splitting on v_i

In order to propose a new cutpoint that is consistent with the entire tree structure, choose a new cutpoint value for c_i from the interval

$$\left(a_i^{v_i}, b_i^{v_i}\right) = \left(max\left(0, min(C^{v_i}_{p(i)}), max(C^{v_i}_{l(i)})\right), min\left(1, max(C^{v_i}_{p(i)}), min(C^{v_i}_{r(i)})\right)\right.$$

Such proposals are entirely consistent with the existing tree structure.

Perturb

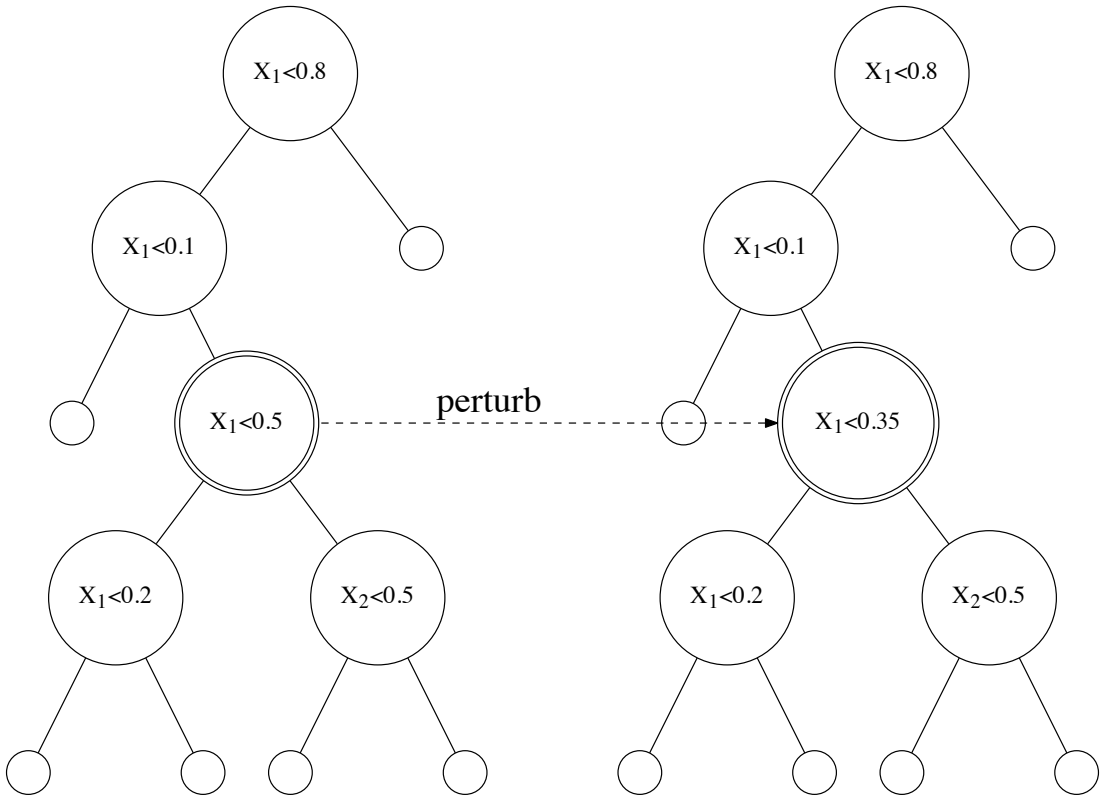
For the perturb MH step, we

1. Generate $c_i' \sim unif(a_i^{v_i}, b_i^{v_i})$
2. Accept c_i' with probability $\alpha = min\left\{1, \frac{\pi(c_i')}{\pi(c_i)}\right\}$

This requires simply computing the ratio of normal likelihoods.

Perturb Example

To perturb at the node $x_1 < 0.5$ (node 5) we have $C^{v1}_{p(5)} = \{0.1, 0.8\}$, $C^{v1}_{l(5)} = \{0.2\}$ and $C^{v1}_{r(5)} = \{\}$.
So, we draw a cutpoint from the range $(0.2, 0.8)$



Perturb within change-of-variable

Previous approaches propose a new variable v_i at node i simply drawing from the prior, giving low acceptance rates
We use a pre-conditioned change-of-variable proposal by proposing changes to variables highly correlated with the existing variable.

$$q(v_k, v_j) = \frac{Cor(X_k, X_j) \times \mathcal{I}_{\left(a_i^{v_j}, b_i^{v_j}\right) \neq \{\}}}{\sum_l Cor(X_k, X_l) \times \mathcal{I}_{\left(a_i^{v_l}, b_i^{v_l}\right) \neq \{\}}}$$

Then, given the new variable, draw a new cutpoint using the perturb procedure described.

Perturb within change-of-variable

For the perturb within change-of-variable MH step, we

1. Generate $v_i' \sim q(v_i, \cdot)$
2. Generate $c_i' \sim unif(a_i^{v_i'}, b_i^{v_i'})$
3. Accept v_i', c_i' with probability $\alpha = min\left\{1, \frac{\pi(v_i', c_i')q(v_i', v_i)}{\pi(v_i, c_i)q(v_i, v_i')}\right\}$

eg: Suppose $p = 3$ and $Cor(X_1, \cdot) = [1.0, 0.0, 0.9]$ and variable 3 has cutpoints available at node 5. Then, a transition from $v_5 = 1 \rightarrow v_5 = 3$ is proposed with probability $\frac{0.9}{1+0.9} \approx 0.47$ and the new cutpoint is drawn from $(a_5^{v_3}, b_5^{v_3}) = (0, 0.7)$.

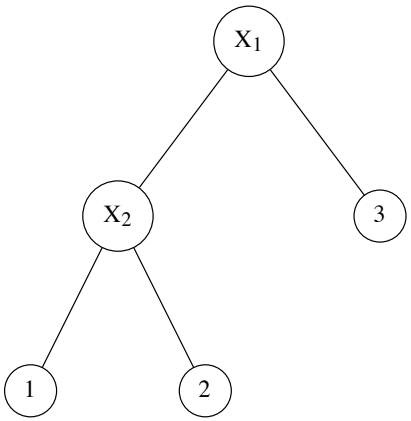
Single Tree Example

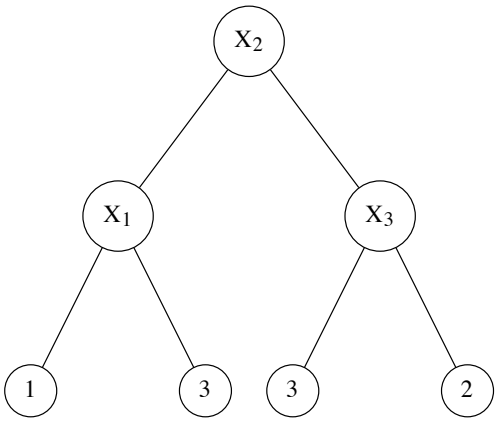
With only b/d, the sampler quickly converged on a single tree representation with 4 terminal nodes (acceptance rate $= 0$)

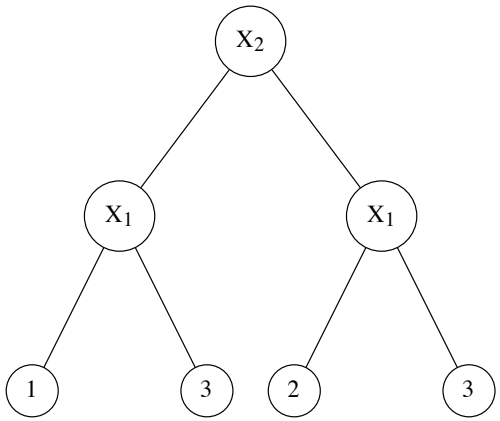
With our modifications included, the MCMC appears to fully sample all trees consistent with the data (acceptance rate $\sim 20\%$)

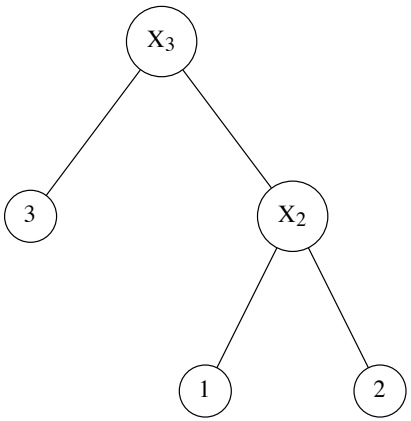
Note that change, swap and restructure proposals do not change tree dimensionality, so even with these proposals it is unlikely the sampler would have found the more parsimoniuous 3-terminal node structure.

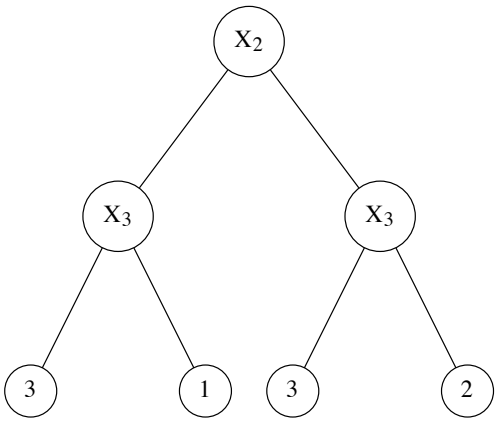
Single Tree Example

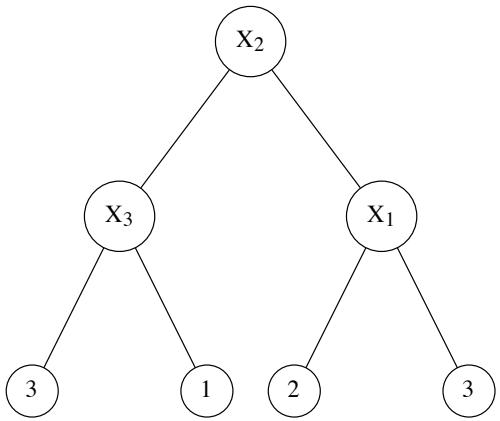








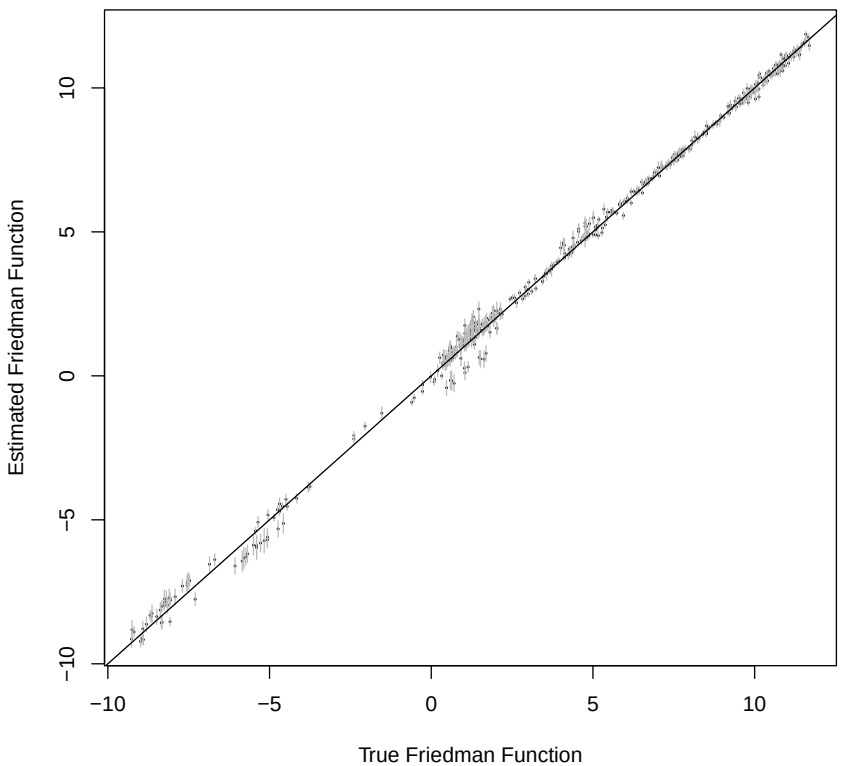


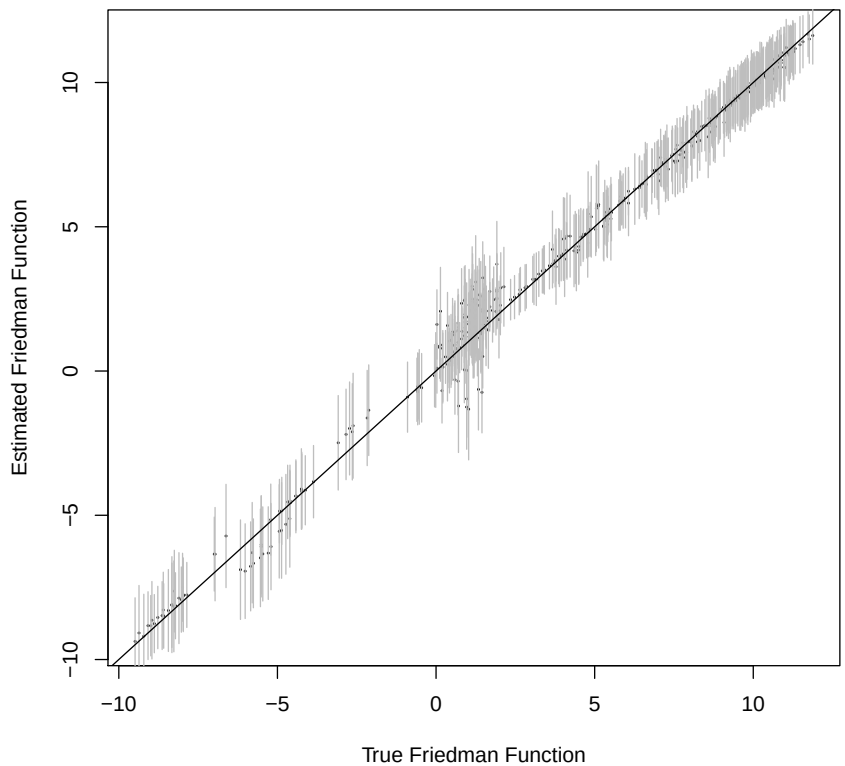


Computer Experiments Example

Fit BART with $m = 200$ trees, acc. rate improves from 4% to 25% or 70%

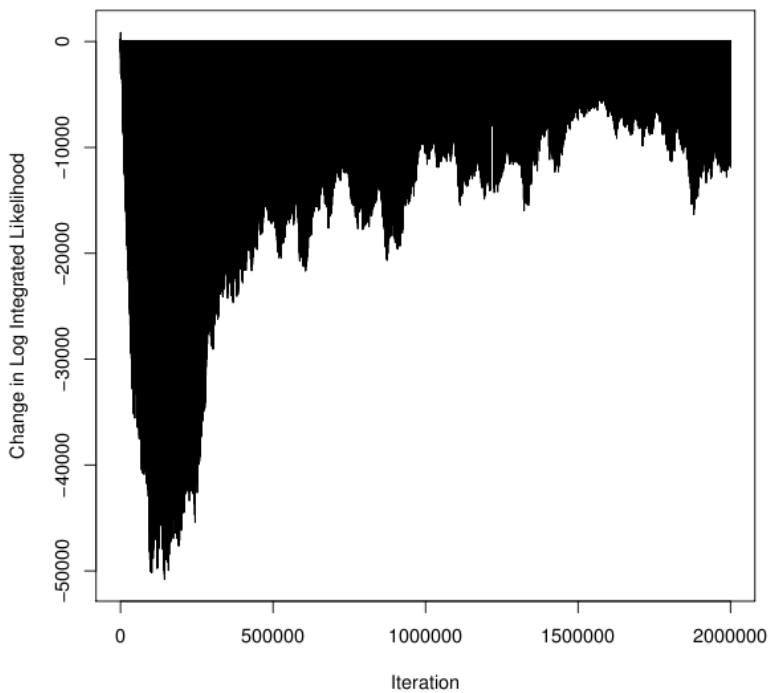
Empirical coverage of the 90% credible interval improves from 53% to 96% or 92%

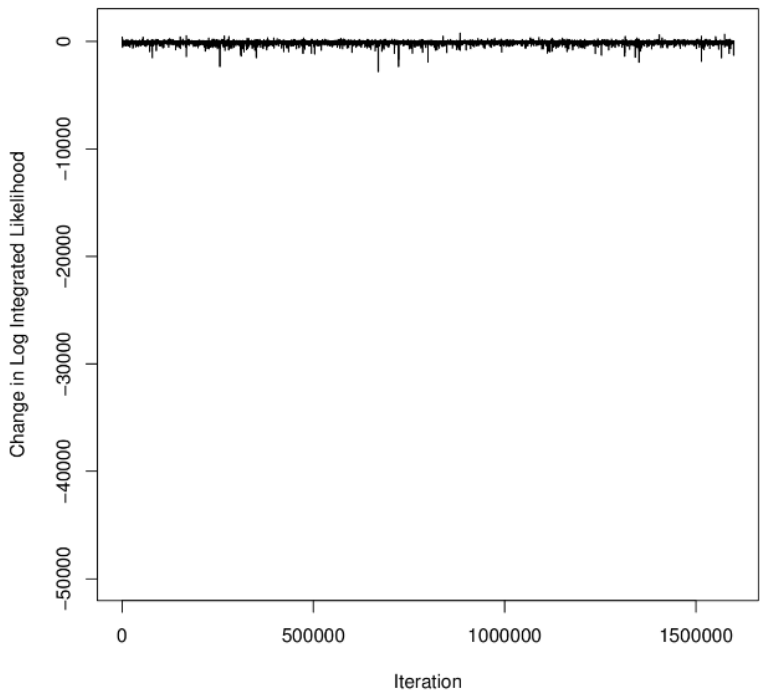




Computer Experiments Example

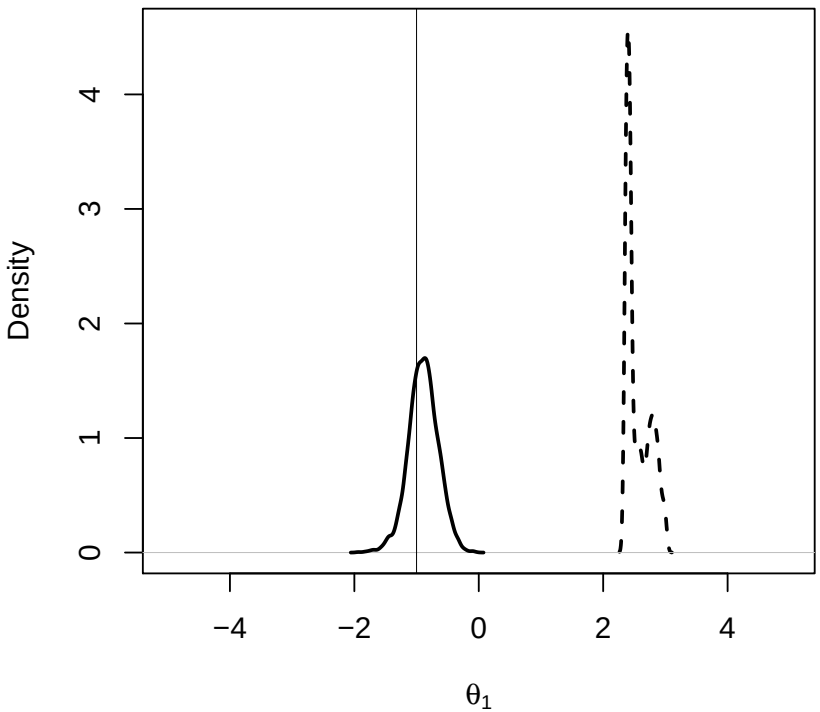
Fit BART with $m = 200$ trees, acc. rate improves from 4% to 25% or 65%
Empirical coverage of the 90% credible interval improves from 53% to 96% or 92%

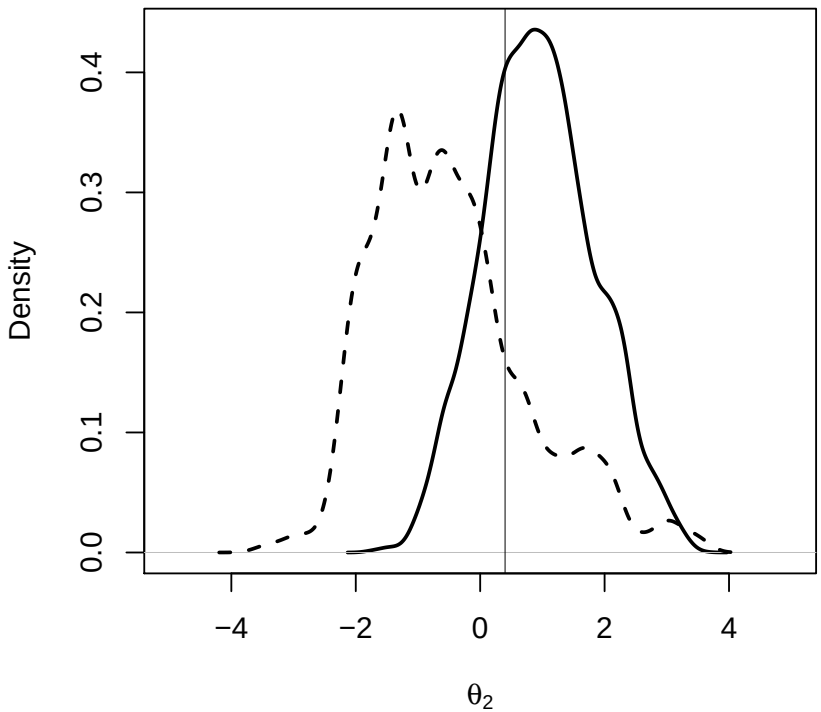




Calibration Example

Regression tree calibration model with and without the proposals developed.
We had $\eta(x,t) = t_1 x + t_2$ and $y_f(x) = -x + 0.4 + 0.1x^2 + \epsilon_f$, so $(\theta_1, \theta_2) = (-1, 0.4)$ and $\delta(x) = 0.1x^2 \sim 0$

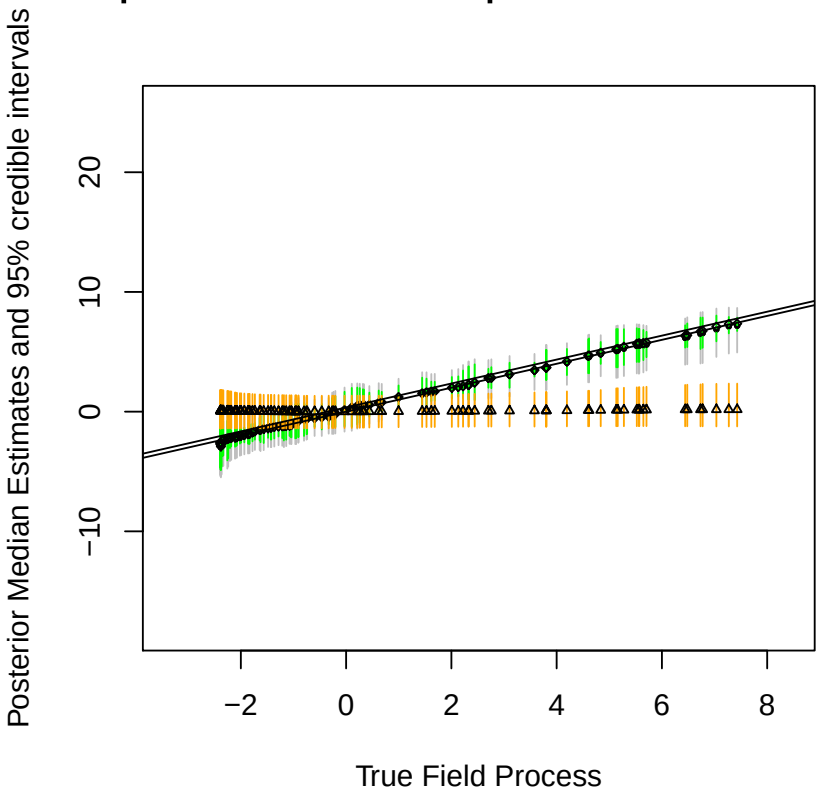


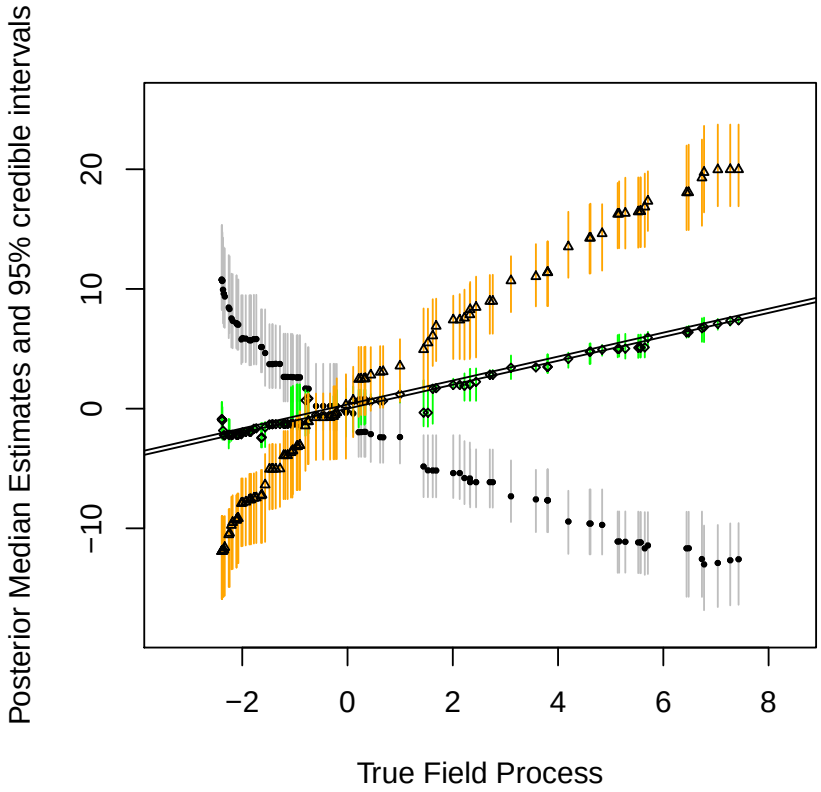


Calibration Example

Regression tree calibration model with and without the proposals developed.

Acceptance rate improved from $<10\%$ to 23%





Conclusion

- Bayesian Regression Tree models have some nice properties which make them well suited to computer experiments problems (non-stationarity, "big data", matrix-free)
- Mixing problematic with small σ^2 or large n - developed two novel MH proposal mechanisms to improve mixing
- The **perturb** proposal efficiently generates proposals that are consistent with the tree
- Extended with a pre-conditioned **change-of-variable** proposal that uses the empirical correlation structure of the covariates
- The **tree rotation** generates dimension-changing proposals at interior nodes of the tree. Efficient since only terminal nodes descendent of the rotation node are needed in computing the integrated likelihood accept/reject step.
- Might be viewed as a swap proposal that retains tree consistency.
- All the proposals developed do not depend on the data.
- Future work with trees & computer experiments: heteroscedasticity, dimension reduction, others...