# BART: Bayesian Additive Regression Trees

Hugh Chipman, Acadia
Edward George, Wharton, U of Pennsylvania
Robert McCulloch, U. of Chicago, Business School

Thanks to Tim Swartz for laying out Bayesian basics.

This is going to be a fully Bayesian approach to
a model built up of many tree models.

We are going to do:

$$f(\theta \mid y, x) \propto f(y \mid x, \theta) f(\theta)$$

where $\theta$ is going to include many tree models.

We have to specify the prior and compute the posterior.

First we need some notation for a single tree model.
We have to be able to think of a tree model as
a "parameter".

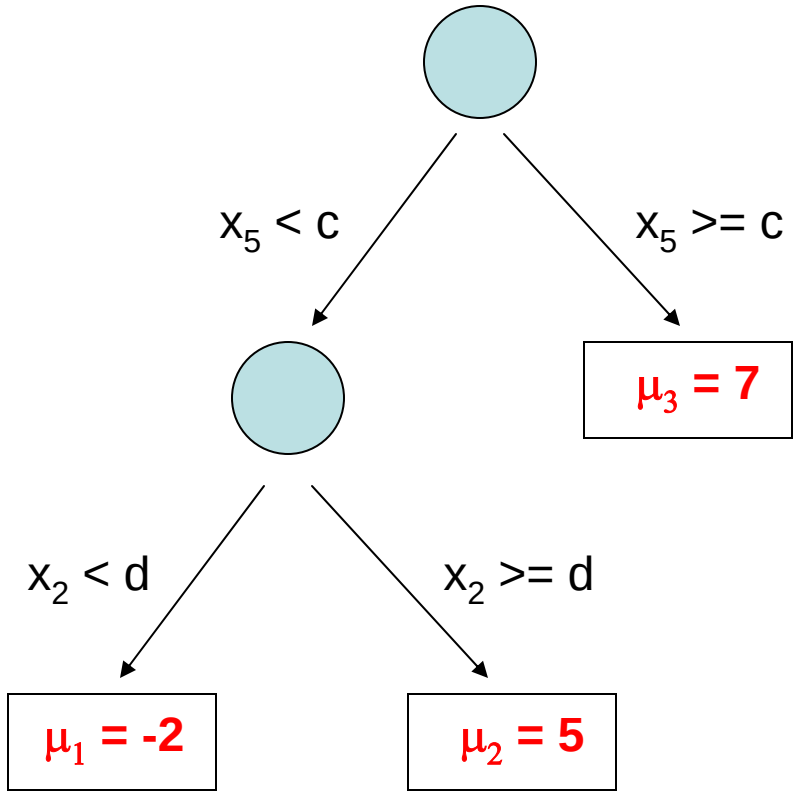
Let T denote the tree structure
including the decision rules.

At bottom node i we have
a parameter $\mu_i$.

Let, $M = \{\mu_1, \mu_2, \ldots, \mu_b\}$

denote the set of $\mu$'s.

g(x,T,M) is then the $\mu$
associated with an x.




Given x, and the parameter value (T,M), g(x,T,M) is our prediction for y.

Let $\{T_j, M_j\}$ be a set of tree models.

Our model is:

$$y = g(x,T_1,M_1) + g(x,T_2,M_2) + \ldots + g(x,T_m,M_m) + \sigma z, \quad z \sim N(0,1)$$

m=hundreds !!! (eg. 200 )

$$\theta = ((T_1,M_1),....(T_m,M_m),\sigma)$$

hundreds of parameters:

only one of which is identified ($\sigma$)

"possibly way too many" - "over complete basis"

this model seems silly, and it is, *if you don't use a lot of prior information !!*

Motivated by "boosting":

overall fit is the sum of many "weak learners"

Prior is key !! :

Prior info: each tree not too big, each $\mu$ not too big,
$\sigma$ could be small

Bayesian Nonparametrics:

Lots of parameters (to make model flexible)
and lots of prior to shrink towards simple structure
(regularize).

Note:

Basic "model space" intuition:
shrinks towards additive models with some interaction.

We'll sketch the MCMC and then lay out the prior.

## Sketch the MCMC

$$y = g(x,T_1,M_1,x) + g(x,T_2,M_2) + \ldots + g(x,T_m,M_m) + \sigma z$$

The "parameter" is $\{T_j\}$, $\{M_j\}$, $\sigma$.

"simple" gibbs sampler:

(1) $\sigma \,|\, \{T_j\}, \{M_j\}, \text{data}$

(2) $(T_j, M_j) \,|\, \{T_i\}_{i \neq j}, \{M_i\}_{i \neq j}, \sigma, \text{data}$ (bayesian backfitting)

(1) subtract all the g's from y and you have a simple problem
(2) subtract all but the $j^{th}$ g from y

The draw

$$(T_j, M_j) \mid \{T_i\}_{i \neq j}, \{M_i\}_{i \neq j}, \sigma, \text{data}$$

is done as

$$p(T,M) = p(T)\, p(M|T)$$

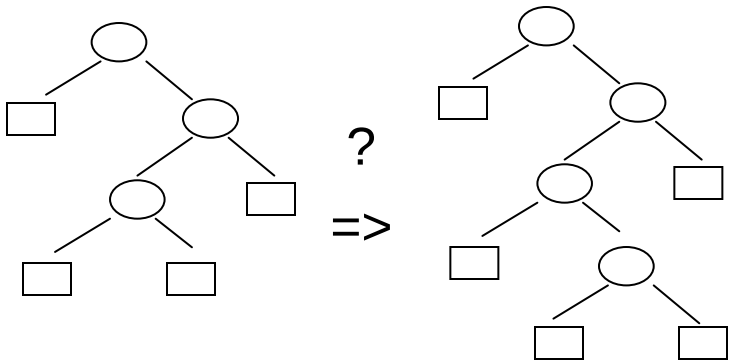
that is, we first margin out M and draw T, then draw M given T.

M|T is easy, just a bunch of normal mean problems
(and we will use the conjugate prior).

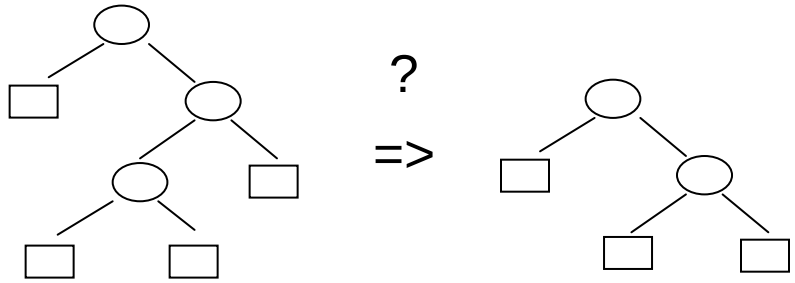
T is drawn using the Metropolis-Hastings algorithm.

Given the current T, we propose a modification and
then either move to the proposal or repeat the old tree.

In particular we have proposals that change the size of the tree:




propose a more complex tree




propose a simpler tree

More complicated models will be accepted if the data's insistence
overcomes the reluctance of the prior.

$$y = g(x,T_1,M_1) + g(x,T_2,M_2) + \ldots + g(x,T_m,M_m) + \sigma\, z, \quad z \sim N(0,1)$$

So, at each iteration, each T, each M and $\sigma$ is updated.

This is a Markov chain such that the stationary distribution is the posterior.

As we run the chain, it is common to observe that an individual tree grows quite large and then collapses back to a single node!!

Each tree contributes a small part to the fit, and the fit is swapped around from tree to tree as the chain runs.

Simulated example:

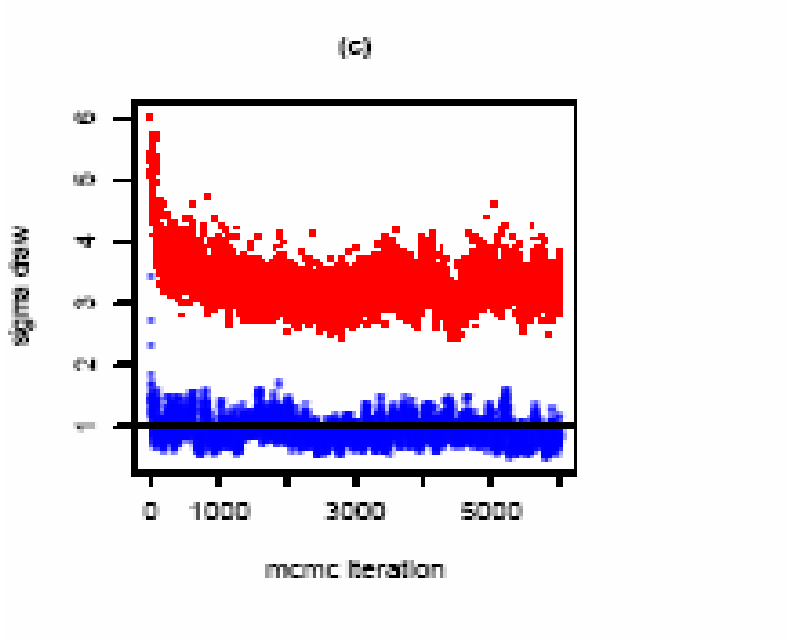
$$y = 10\sin(\pi x_1 x_2) + 20(x_3 - .5)^2 + 10x_4 + 5x_5 + 0x_6 + \cdots 0x_{10} + \sigma Z$$

used by Friedman, n=100, $\sigma$ = 1.

Blue is draws of $\sigma$ with 200 trees.

Draws quickly burn-in and then vary about the true value.

Red is with m=1.



## The Prior

Make everything you can independent.
But prior on M must be conditional on the corresponding T because the dimension is the number of bottom nodes.

$$p((T_1, M_1), (T_2, M_2), \ldots, (T_m, M_m), \sigma)$$

$$= p(T_1, T_2, \ldots, T_m)\, p(M_1, M_2, \ldots, M_m \,|\, T_1, T_2, \ldots, T_m)\, p(\sigma). \tag{5}$$

Since the dimension of each $M_j$ depends on the corresponding $T_j$ this conditional structure is essential. We simplify further by imposing independence whenever possible:

$$p(T_1, T_2, \ldots, T_m) = \prod p(T_j), \tag{6}$$

$$p(M_1, M_2, \ldots, M_m \,|\, T_1, T_2, \ldots, T_m) = \prod p(M_j \,|\, T_j), \tag{7}$$

$$p(M_j \,|\, T_j) = \prod p(\mu_{ij} \,|\, T_j), \tag{8}$$

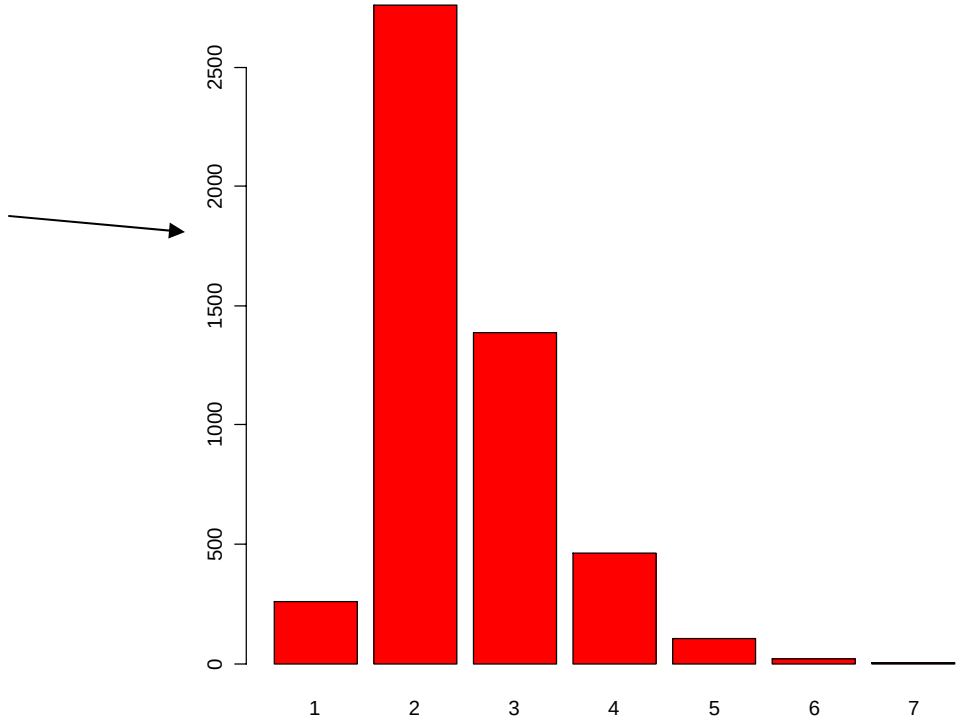
So just choose p(T), p($\sigma$), and p($\mu$|T)=p($\mu$)

## p(T)

Not obvious. We specify a process that grows trees.

Marginal prior on number of bottom nodes.

*Put prior weight on small trees!!*




There are parameters associated with this prior but we have not played with them at all in BART.

## p(μ|T)

First we standardize y so that with high probability E(Y|x) is in the interval [-.5,.5].

choose: $\mu \sim N(0, \sigma_\mu^2)$

In our model, E(Y|x) is the sum of m indepedent μ's (a priori).

So the prior standard deviation of E(Y|x) is $\sqrt{m}\sigma_\mu$

Let $k\sqrt{m}\sigma_\mu = .5 \Rightarrow \sigma_\mu = \frac{.5}{k\sqrt{m}}$

k is the number of standard deviations from the mean of 0 to the interval boundary of .5

This is a knob. Our default is k=2.

## p(σ)

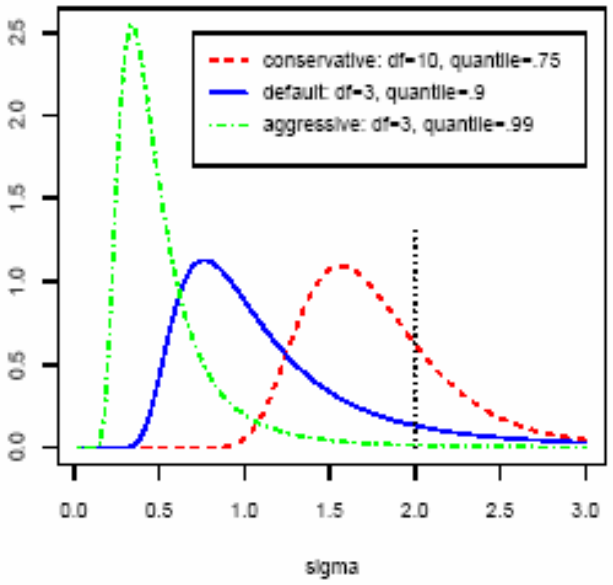
First choose "rough estimate" $\hat{\sigma}$ (least squares estimate, sd(y), just choose it)

Let $\sigma^2 \sim \frac{\nu\lambda}{\chi^2_\nu}$

and choose $\nu$ and then a quantile to put the rought estimate at (this determines $\lambda$)

Here are the three priors we have been using:

$\hat{\sigma} = 2$



## Prior summary:

We have fixed the prior on T.

For m, just have k. Default is 2, might try 3.

Three priors for $\sigma$, given rough estimate.

Not to many knobs and there are simple default recommendations!!

Have to standardize y and x's, but standardization of x not is sensitive an issue as in say neural nets.

Claim: it is easy to use.

In practice, we do use the data to pick the prior, but you could easily just choose it.

Note:

At iteration i we have a draw from the posterior of the function

$$f_i(\bullet) = g(\bullet, T_{1i}, M_{1i}) + g(\bullet, T_{2i}, M_{2i}) + \cdots + g(\bullet, T_{mi}, M_{mi})$$

Think of f as a "parameter" and we are drawing from its posterior.

To get in-sample fits we average $f_i(x)$ for an x in-sample.

Similarly, we can get out-of-sample fits for out-of-sample x's.

Posterior uncertainty about f(x) is captured by the set of draws $f_i(x)$.

Combines boosting "ensemble learning" with
Bayesian model averaging.

# Friedman Simulated Example

$$y = 10\sin(\pi x_1 x_2) + 20(x_3 - .5)^2 + 10x_4 + 5x_5 + 0x_6 + \cdots 0x_{10} + \varepsilon$$

10 x's, only first 5 matter.

Compare with other fitting techniques
(Neural nets,Random forests, boosting, MARS, linear regression)

- 50 simulation of 100 observations
- 10 fold cross validation used to pick tuning parameters, then refit will all 100

Performance measured by:

$$\text{RMSE} = \sqrt{\frac{1}{1000}\sum_{i=1}^{1000}(\hat{f}(x_i) - f(x_i))^2}$$

where x's are 1000 out of sample draws

10 fold cross validation is used to pick tuning parameters.

BART-cv uses cv to choose prior setting

BART-default just goes with a single prior choice

| Method | average RMSE | se(RMSE) |
|---|---|---|
| Random Forests | 2.655 | 0.025 |
| Linear Regression | 2.618 | 0.016 |
| Neural Nets | 2.156 | 0.025 |
| Boosting | 2.013 | 0.024 |
| MARS | 2.003 | 0.060 |
| BART-cv | 1.787 | 0.021 |
| BART-default | 1.759 | 0.019 |

*have lots of examples where BART does great out of sample !!!!!!!*

Took Friedman example and added more useless x's

*Fit BART with*
*1000 x's*
*and only*
*100 observations*
*and got*

*"reasonable"*

*results !!!!*

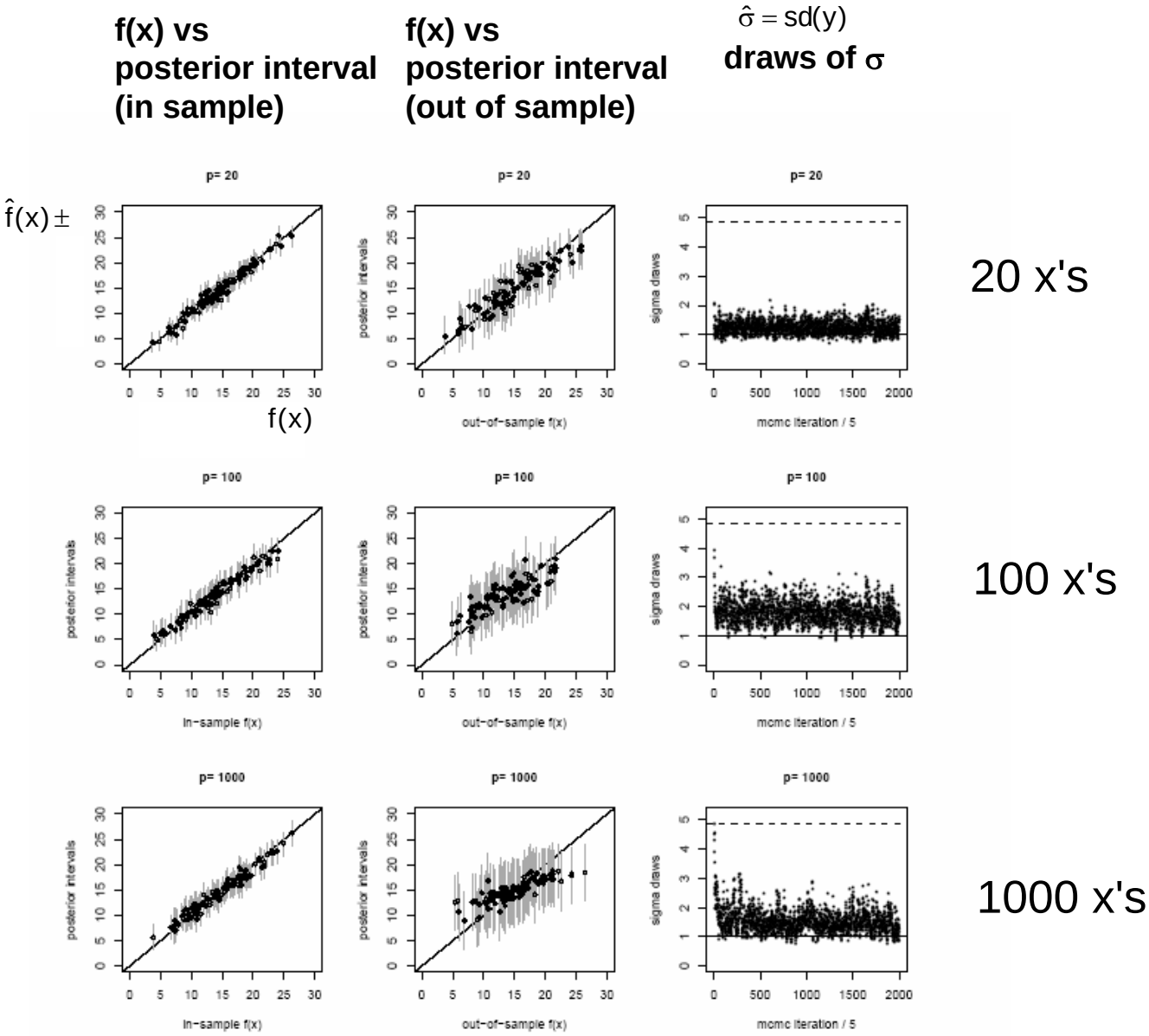



Figure 4: Inference about Friedman's function in $p$ dimensions.

## Things I like about BART:

Competitive out of sample performance
(mcmc stochastic search (birth and death), boosting, model averaging)

Simplicity of underlying tree model leads to simple prior.
(have used same prior with 1 x as with 1,000!)

Easy to use! (again, because of prior, have R package)

Stable, run twice get same thing
Converges quickly
Mixes reasonably well.
(intuition, as you run it, individual trees grow
and then shrink back to nothing)

Posterior uncertainty (relative to other "data mining" tools).

## Hockey Example (with Jason Abrevaya)

Abrevaya and McCulloch, “Reversal of Fortune”

Theory:

NHL hockey is impossible to officiate (fast, tradition of violence)
Hence, ***refs will make calls even out***.

Ken Hitchcock:
"there could probably be a penalty called on every NHL shift"

Glen Healy:

"Referees are predictable.The flames have had three penalties,
I guarantee you the oilers will have three."

If ref calls too many penalties:

“Let the players play!!”

If he calls too few:

“Hey ref, get control of the game"

Table 1: Distribution of Penalty Types

| Penalty Type | Frequency |
|---|---|
| Roughing | 15.93% |
| Holding | 12.04% |
| Hooking | 11.20% |
| Fighting | 10.99% |
| Interference | 10.18% |
| Tripping | 8.69% |
| High sticking | 7.72% |
| Slashing | 6.85% |
| Cross checking | 4.96% |
| Unsportsmanlike conduct | 2.45% |
| Elbowing | 2.07% |
| Boarding | 1.91% |
| Too many men on ice | 1.20% |
| Goalie interference | 1.17% |
| Charging | 1.03% |
| Delay of game | 0.72% |
| Diving | 0.17% |
| Spearing | 0.10% |
| Board Check | 0.10% |
| Butt ending | 0.01% |
| Attempt to injure | 0.01% |

Have data on every penalty called in the NHL from 1995 to 2001.
57,883 observations.

y = 0 if pen on same team as last time, 1 else

$$\bar{y} = .589$$

59% of the time, the call reverses.

Table 5: Variable Descriptions

| Variable | Description | Mean | Min | Max |
|---|---|---|---|---|
| *Dependent variable* | | | | |
| `revcall` | 1 if current penalty and last penalty are on different teams | 0.589 | 0 | 1 |
| *Indicator-Variable Covariates* | | | | |
| `ppgoal` | 1 if last penalty resulted in a power-play goal | 0.157 | 0 | 1 |
| `home` | 1 if last penalty was called on the home team | 0.483 | 0 | 1 |
| `inrow2` | 1 if last two penalties called on the same team | 0.354 | 0 | 1 |
| `inrow3` | 1 if last three penalties called on the same team | 0.107 | 0 | 1 |
| `inrow4` | 1 if last four penalties called on the same team | 0.027 | 0 | 1 |
| `tworef` | 1 if game is officiated by two referees | 0.414 | 0 | 1 |
| *Categorical-variable covariate* | | | | |
| `season` | Season that game is played (e.g., 1995 for 95-6 season) | | 1995 | 2001 |
| *Other covariates* | | | | |
| `timeingame` | Time in the game (in minutes) | 31.44 | 0.43 | 59.98 |
| `dayofseason` | Number of days since season began | 95.95 | 1 | 201 |
| `numpen` | Number of penalties called so far (in the game) | 5.76 | 2 | 21 |
| `timebetpens` | Time (in minutes) since the last penalty call | 5.96 | 0.02 | 55.13 |
| `goaldiff` | Goals for last penalized team minus goals for opponent | -0.02 | -10 | 10 |
| `gf1` | Goals/game scored by the last team penalized | 2.78 | 1.84 | 4.40 |
| `ga1` | Goals/game allowed by the last team penalized | 2.75 | 1.98 | 4.44 |
| `pf1` | Penalties/game committed by the last team penalized | 6.01 | 4.11 | 8.37 |
| `pa1` | Penalties/game by opponents of the last team penalized | 5.97 | 4.33 | 8.25 |
| `gf2` | Goals/game scored by other team (not just penalized) | 2.78 | 1.84 | 4.40 |
| `ga2` | Goals/game allowed by other team | 2.78 | 1.98 | 4.44 |
| `pf2` | Penalties/game committed by other team | 5.96 | 4.11 | 8.37 |
| `pa2` | Penalties/game by opponents of other team | 5.98 | 4.33 | 8.25 |

There are a lot of descriptive statistics in the paper.

Goal of the study:

Which variables have an "important effect " on y?
(In particular the "inrows")

Fit BART.
$y = p(x) + \varepsilon$
Again outperformed competitors.

How can we explore the BART fit to see what it has to tell us ?

We picked a subset of 4 factors and did a 2^4 experiment.
All the other variables are set at a base setting.

Table 7: Description of Scenario Design

| Quantity | Code | Meaning | Code | Meaning |
|---|---|---|---|---|
| goal differential | $g$ | `goaldiff` $= -1$ *(last penalized team behind by a goal)* | $G$ | `goaldiff` $= 1$ *(last penalized team ahead by a goal)* |
| consecutive calls | $r$ | `inrow2` $= 0$ *(last two calls on different teams)* | $R$ | `inrow2` $= 1$ *(last two calls on same team)* |
| time between penalties | $t$ | `timebetpens` $= 2$ *(short time since last penalty)* | $T$ | `timebetpens` $= 7$ *(long time since last penalty)* |
| time in the game | $n$ | `timeingame` $= 10$, `numpens` $= 3$ *(early in the game)* | $N$ | `timeingame` $= 55$, `numpens` $= 12$ *(late in the game)* |

So, gRtn, means:

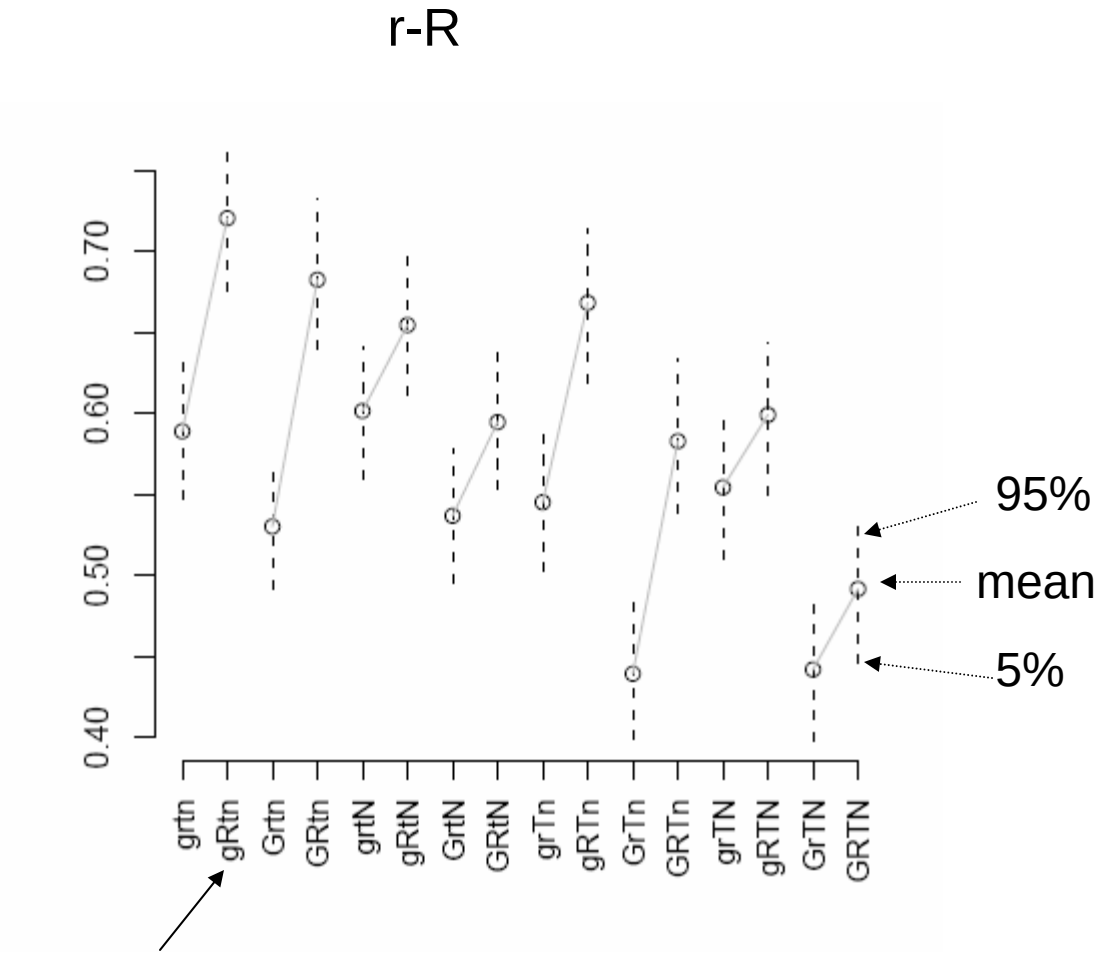
g: the last penalized team down by 1
R: last two calls on same team
t: not long since last call
n: early in the game

We have 16 possible x configurations.

Report the posterior of p(x) at each x, where p is the random variable.

Huge amount of "significant" fit.

Interaction.




**last penalized team:**
**down by a goal**
**had last two pens, not long ago**
**early in the game**

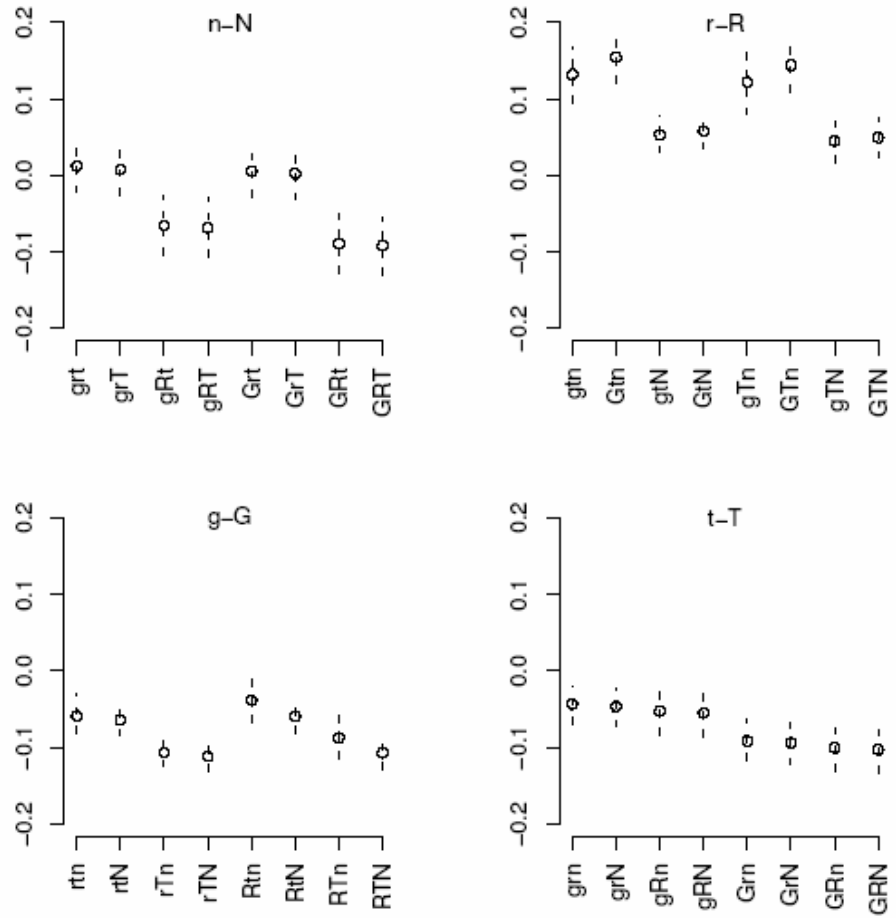



Figure 11: *BART inference for interaction scenarios, differences.*

posterior of differences from previous slide

Posterior of

p(x,R)-p(x,r)

at 8 possible x.

Other three plots are for the other three factors.

Google Robert McCulloch

R instructions for Linux and Windows
(soon on CRAN)

I'll put up a "main" to run outside of R.