Beyond axis-alignment:
BART w/ categorical variables & oblique decision rules

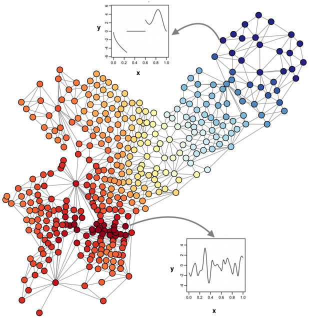
Sameer K. Deshpande

University of Wisconsin–Madison

19 October 2023

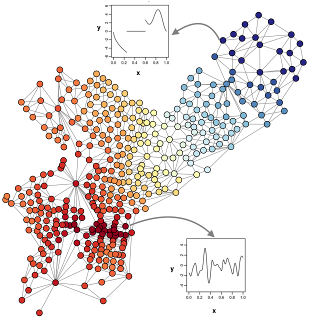
Nonparametric network-linked regression

- Observe data on a network $\mathcal{G} = (V, E)$
- n_v pairs $(\boldsymbol{x}_{vt}, y_{vt})$ at vertex v
- $\mathbb{E}[y|\boldsymbol{x}]$ may vary across network
- I.e. $\boldsymbol{x} \in [0,1]^p$ *interacts* w/ v



Nonparametric network-linked regression

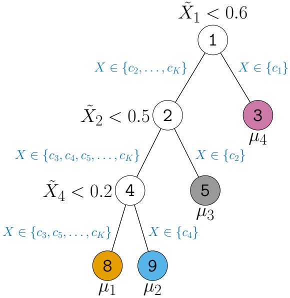
- Observe data on a network $\mathcal{G} = (V, E)$
- n_v pairs $(\boldsymbol{x}_{vt}, y_{vt})$ at vertex v
- $\mathbb{E}[y|\boldsymbol{x}]$ may vary across network
- I.e. $\boldsymbol{x} \in [0,1]^p$ *interacts* w/ v



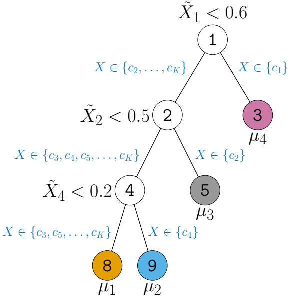
- Idea: introduce vertex label v as covariate
- Model: $y_{vt} \sim \mathcal{N}(f(\boldsymbol{x}_{vt}, v), \sigma^2)$
- Hard to pre-specify correct functional form
- Encourage network smoothness so $f(\boldsymbol{x}, v) \approx f(\boldsymbol{x}, v')$ if $v \sim v'$?

Pitfalls of one-hot encoding

- $X \in \{c_1, \ldots, c_K\}$
- One-hot encoding: $X \to (\tilde{X}_1, \ldots, \tilde{X}_K)$
  - $\tilde{X}_k = \mathbb{1}(X = c_k)$
  - Treat $\tilde{X}_k$'s as continuous



Pitfalls of one-hot encoding



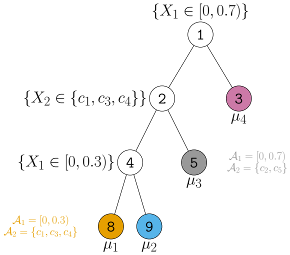
- $X \in \{c_1, \ldots, c_K\}$
- One-hot encoding: $X \to (\tilde{X}_1, \ldots, \tilde{X}_K)$
  - $\tilde{X}_k = \mathbb{1}(X = c_k)$
  - Treat $\tilde{X}_k$'s as continuous
- Decision trees built w/ $\tilde{X}_k$'s can induce $2^K - K$ partitions of the form

$$\overbrace{\{c_1\} \cup \cdots \{c_k\}}^{k \text{ singletons}} \cup \overbrace{\{c_{k+1}, \ldots, c_K\}}^{\text{set with } K-k \text{ elements}}$$

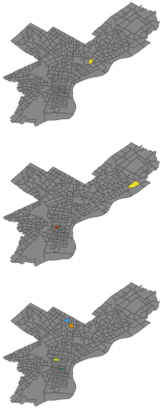
- Bell number $B_k \gg (K/2)^{K/2} \gg 2^K - K$
- Zero prior probability on overwhelming majority of partitions of levels
- One-hot encoding can't leverage network structure...

Decision rule prior

1. Draw $j \sim \text{Multinomial}(\theta_1, \ldots, \theta_p)$ where $\theta_j = \mathbb{P}(\text{split on } X_j)$
2. Compute set of all available values $\mathcal{A}_j$
    - $\mathcal{A}_j$ determined by rules at ancestors
    - X_j continuous $\rightarrow \mathcal{A}$ is an interval
    - X_j categorical $\rightarrow \mathcal{A}$ is discrete set
3. Draw random subset $\mathcal{C}$ from $\mathcal{A}_j$
    - X_j continuous: set $\mathcal{C} = [0, c)$, $c \sim \mathcal{U}(\mathcal{A}_j)$
    - X_j categorical: assign elements of $\mathcal{A}_j$ to $\mathcal{C}$ with probability 0.5

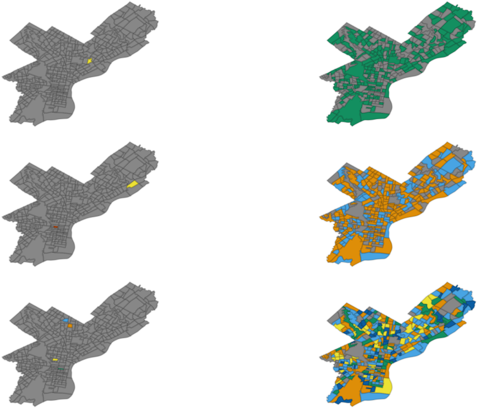


Example: Partitions of Philadelphia's census tracts



☹ One-hot encoding: no "borrowing strength" across tracts

Example: Partitions of Philadelphia's census tracts

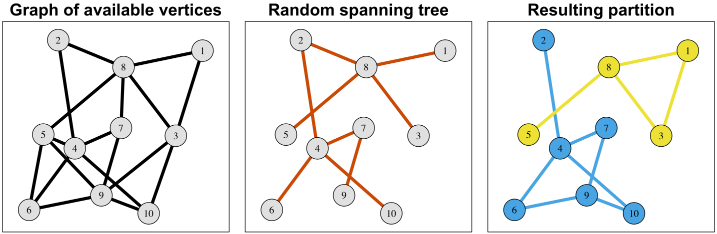


- ☹ One-hot encoding: no "borrowing strength" across tracts
- ☺ Uniformly partitioning levels: pools many tracts together . . .
- ☹ . . . but partially pools data across geographically disparate regions

Example: Partitions of Philadelphia's census tracts

- ☹ One-hot encoding: no "borrowing strength" across tracts
- ☺ Uniformly partitioning levels: pools many tracts together ...
- ☹ ... but partially pools data across geographically disparate regions
- How can we leverage adjacency and achieve spatial smoothness???

Cutting spanning trees



- Goal: partition $\mathcal{G}[\mathcal{A}]$ into two connected components
    1. Draw uniform spanning tree of $\mathcal{G}[\mathcal{A}]$ (Wilson's algorithm)
    2. Delete an edge from spanning tree to partition into two components

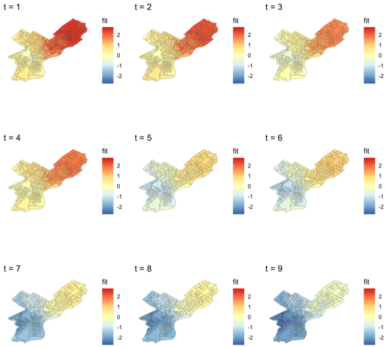
☺ Full support: positive prior prob. on every partition of $\mathcal{G}$ into connected components

Example: prior draws of $f(v, t)$

- Conjecture: infinite-tree limit is Gaussian process $\mathrm{GP}(0, k)$

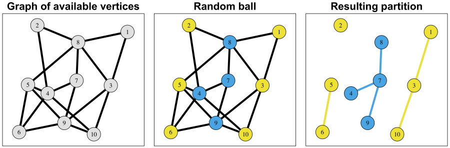
  $k((\boldsymbol{x}, v), (\boldsymbol{x}', v')) \propto$
  $\mathbb{P}((\boldsymbol{x}, v)$ and $(\boldsymbol{x}', v')$ in same leaf$)$

- k determined implicitly by a *recursive partitioning process*



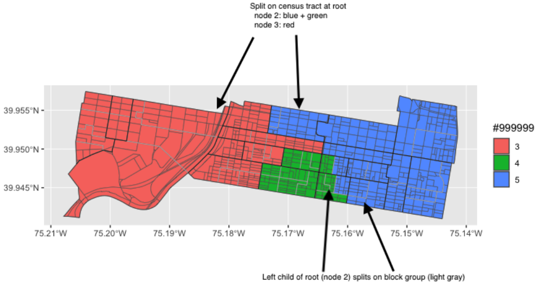
Prior draw of $M = 100$ regression tree ensemble

Other ways to partition networks



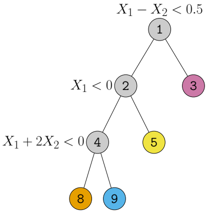
- 😑 Deleting uniform edge from spanning tree can create singletons
- Idea 1: form "hot spots" based on a random ball
  - ☺ Simpler to implement: pick a vertex & take all within random distance
  - ☹ Does not maintain connectivity
- Idea 2: use second smallest eigenvector of graph Laplacian
- Network partitioning processes determines degree of smoothness
  - Conjecture: infinite tree limit is a GP w/ kernel k
  - $k(v, v') = \mathbb{P}(v \text{ \& } v'$ co-clustered in random recursive partitioning).

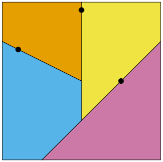
Multiple resolutions



- Idea: run BART w/ additional network-structured predictors
- Challenge: how strictly do we enforce the hierarchical structure?
  - Split on high resolution after exhausting all lower resolutions?
  - Allow splits on lower resolution after splitting on higher resolution?

Oblique BART





- Categorical rule $\{X_j \in \mathcal{C}\}$ equivalent to $\{\sum_{c \in \mathcal{C}} \mathbb{1}(X_j = c) < 0.5\}$
- Oblique trees: decision rules based on *linear combinations* of x_j's
- Implementation: easier to use rules of form $\{\phi^\top(\boldsymbol{x} - \boldsymbol{x}_0) < 0\}$
- $\boldsymbol{x}_0$ drawn from polytope and normal direction ϕ draw from a prior

Parting thoughts & other on-going projects

- Don't one-hot encode categorical predictors w/ regression trees!
- BART with random basis expansion in leafs for interrupted time series
- BART with image input
- Extending BART to more complex input space involves implementing
  - Recipe to randomly split $X \subseteq \mathcal{X}$ into two disjoint pieces
  - Efficient representation of subsets of $\mathcal{X}$
  - Fast way to check whether $\boldsymbol{x}$ is in a given subset

Parting thoughts & other on-going projects

- Don't one-hot encode categorical predictors w/ regression trees!
- BART with random basis expansion in leafs for interrupted time series
- BART with image input
- Extending BART to more complex input space involves implementing
  - Recipe to randomly split $X \subseteq \mathcal{X}$ into two disjoint pieces
  - Efficient representation of subsets of $\mathcal{X}$
  - Fast way to check whether $\boldsymbol{x}$ is in a given subset

Thanks, y'all!

Email: sameer.deshpande@wisc.edu
Package: https://github.com/skdeshpande91/flexBART
Website: https://skdeshpande91.github.io
Preprint: [arXiv:2211.04456]