An Intro. to Tree based models

Santiago Olivella

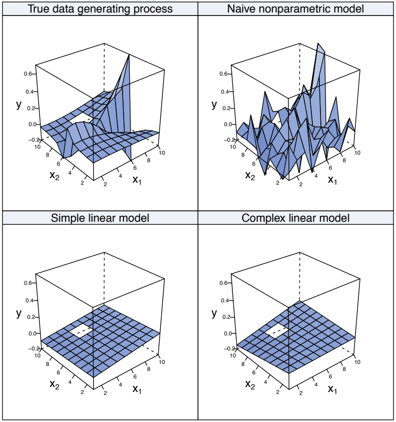
UNC-CH Political Science

July 23, 2020

Motivation: Curse of dimensionality and model dependence

- Covariate spaces are generally not densely populated: curse of dimensionality will affect even Big Data at the largest scale.
- A common solution is to rely on strong assumptions about the DGP, particularly in terms of functional form definition:
  - Ease of interpretation...
  - ...steep price in terms of accuracy.
- **High model dependency**, and large "researcher degrees of freedom".
  - Unless strong theory justifies these modeling assumptions, they may be too restrictive/leave too much room for researcher intervention.

Motivation: A simple example



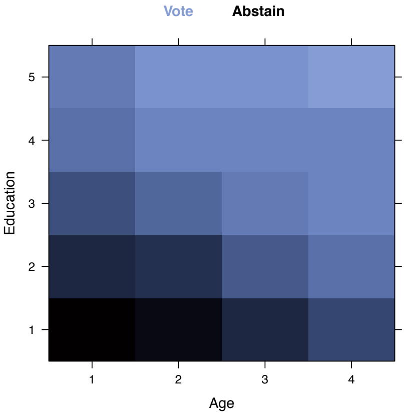
- Flexible models + regularizing mechanisms - researcher intervention
  - That's what **tree ensembles** offer.

Outline

- Tree based methods: an introduction
- A few applications
    - Small-group preference estimation
    - Propensity score estimation
    - Durverger's Law
- Conclusion

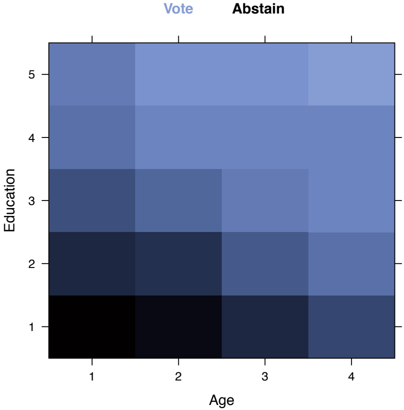
Single tree models: CART

- How to improve null prediction using covariate values?



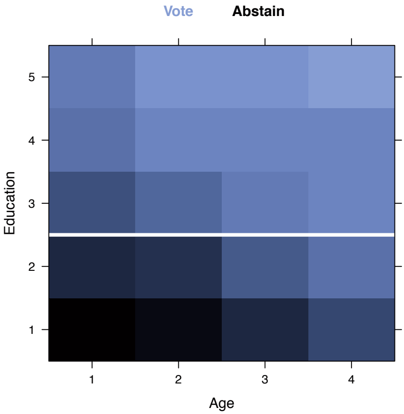
Single tree models: CART

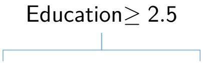
- How to improve null prediction using covariate values?
- Find homogeneous regions in covariate space, and predict within them.



Single tree models: CART

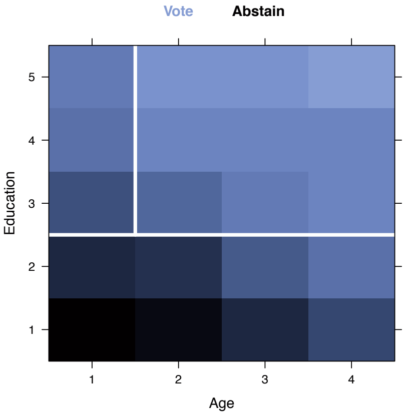
- How to improve null prediction using covariate values?
- Find homogeneous regions in covariate space, and predict within them.

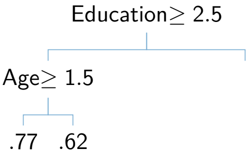



Education≥ 2.5

Single tree models: CART

- How to improve null prediction using covariate values?
- Find homogeneous regions in covariate space, and predict within them.





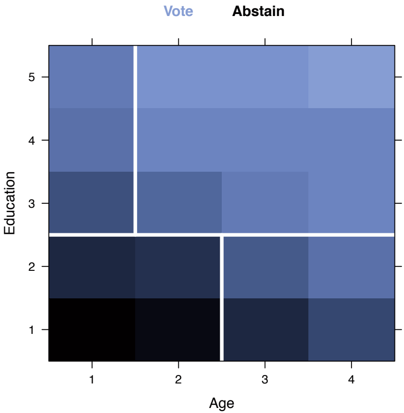
Education≥ 2.5

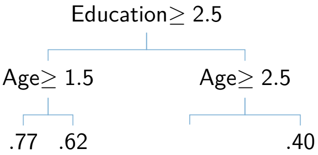
Age≥ 1.5

.77 .62

Single tree models: CART

- How to improve null prediction using covariate values?
- Find homogeneous regions in covariate space, and predict within them.





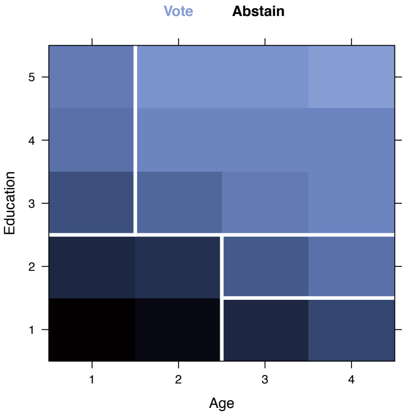
Education$\geq$ 2.5

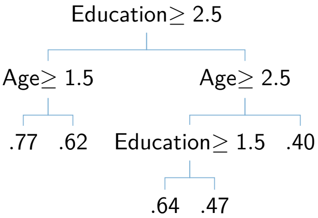
Age$\geq$ 1.5 Age$\geq$ 2.5

.77 .62 .40

Single tree models: CART

- How to improve null prediction using covariate values?
- Find homogeneous regions in covariate space, and predict within them.





Education≥ 2.5

Age≥ 1.5

Age≥ 2.5

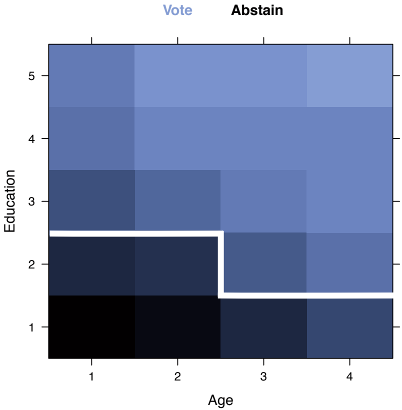
.77 .62

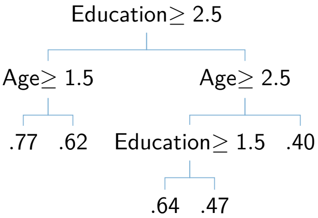
Education≥ 1.5 .40

.64 .47

Single tree models: CART

- How to improve null prediction using covariate values?
- Find homogeneous regions in covariate space, and predict within them.





Education≥ 2.5

Age≥ 1.5

Age≥ 2.5

.77 .62

Education≥ 1.5

.40

.64 .47

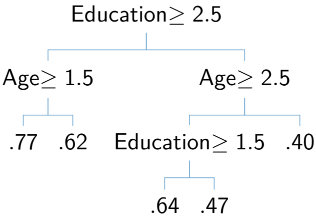
Single tree models: CART

$$f(\mathbf{x}_i) = T(\mathbf{x}_i;\Theta) \equiv \sum_{b=1}^{B} \hat{y}_b \mathbb{1}(\mathbf{x}_i \in R_b)$$

Where Θ defines the splitting rules and terminal node values, and $\mathbb{1}$ is the indicator function.

The goal: find

$$\arg\min_{\Theta} L(f(\mathbf{x_i}), y_i; \Theta)$$



CART: Recursive binary splitting

	Purpose	Description
1	Calculate optimal splits	For each covariate j, calculate the optimal point (v) to create a new split.
2	Choose optimal covariate	Select the covariate and split rule that minimize $L(\cdot)$ using the average y_i in the corresponding regions as c_b.
3	Check stopping rules for new leaves	Check whether the tree has reached pre-specified level of complexity.
4	Repeat steps 1-3	For each new leaf, if the stopping rule has not been reached, add a new split.

CART: pruning

$$f(X_i) = T(X_i;\Theta) \equiv \sum_{b=1}^{B} c_b I(X_i \in R_b)$$

$$\hat{\Theta} = \arg\min_{\Theta} \sum_{b=1}^{B} \sum_{X_i \in R_b} L(y_i, c_b)$$

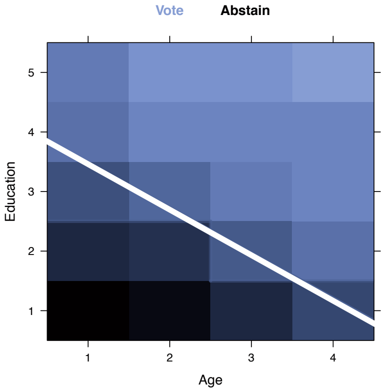
- Prune by:
  - Find subtree T that minimizes $C_\alpha(T) = \sum_{b=1}^{B} \sum_{X_i \in R_b} L(y_{i:X_i \in R_b}, c_b) + \alpha B$
  - B is the number of terminal nodes
  - $\alpha \geq 0$ is user specified

Issues with CART

- Single-tree models are known to be poor predictors

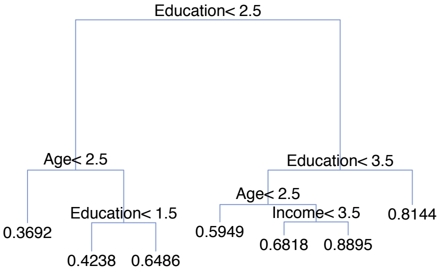
Issues with CART

- Single-tree models are known to be poor predictors
    - Difficulty picking up on additive/linear relations.



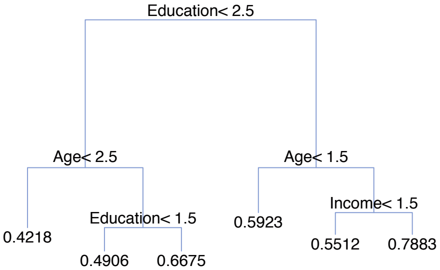
Issues with CART

- Single-tree models are known to be poor predictors
  - Difficulty picking up on additive/linear relations.
  - Very sensitive to changes in data.



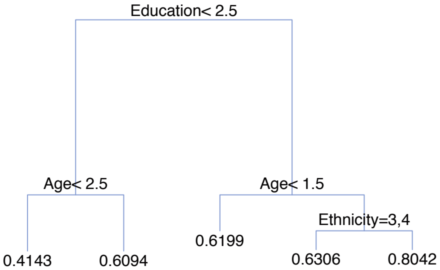
Issues with CART

- Single-tree models are known to be poor predictors
  - Difficulty picking up on additive/linear relations.
  - Very sensitive to changes in data.



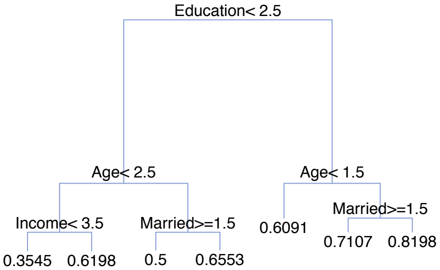
Issues with CART

- Single-tree models are known to be poor predictors
    - Difficulty picking up on additive/linear relations.
    - Very sensitive to changes in data.



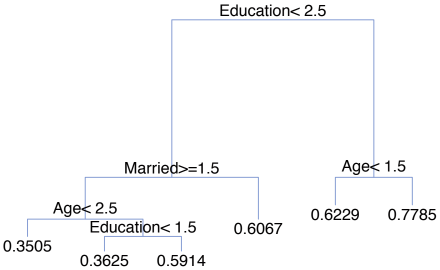
Issues with CART

- Single-tree models are known to be poor predictors
  - Difficulty picking up on additive/linear relations.
  - Very sensitive to changes in data.



Issues with CART

- Single-tree models are known to be poor predictors
  - Difficulty picking up on additive/linear relations.
  - Very sensitive to changes in data.



Issues with CART

- Single-tree models are known to be poor predictors
    - Difficulty picking up on additive/linear relations.
    - Very sensitive to changes in data.
    - No "natural" measure of uncertainty.

Issues with CART

- Single-tree models are known to be poor predictors
    - Difficulty picking up on additive/linear relations.
    - Very sensitive to changes in data.
    - No “natural” measure of uncertainty.

- **Solution**: Ensemble models overcome these issues by:

Issues with CART

- Single-tree models are known to be poor predictors
  - Difficulty picking up on additive/linear relations.
  - Very sensitive to changes in data.
  - No “natural” measure of uncertainty.

- **Solution**: Ensemble models overcome these issues by:
  - Fitting many individual “weak” trees

Issues with CART

- Single-tree models are known to be poor predictors
    - Difficulty picking up on additive/linear relations.
    - Very sensitive to changes in data.
    - No “natural” measure of uncertainty.

- **Solution**: Ensemble models overcome these issues by:
    - Fitting many individual “weak” trees
    - Combining their predictions in a weighted average

Issues with CART

- Single-tree models are known to be poor predictors
  - Difficulty picking up on additive/linear relations.
  - Very sensitive to changes in data.
  - No “natural” measure of uncertainty.

- **Solution**: Ensemble models overcome these issues by:
  - Fitting many individual “weak” trees
  - Combining their predictions in a weighted average
  - Embedding trees in probabilistic model.

Single tree models: recap

- Pros:
  - Intuitive!
  - Handles lots of data issues easily (e.g. different measurement levels, different scales & missing values).
  - Great at modeling and identifying relevant interactions.
  - Automatic feature selection.
- Cons:
  - Performs poorly with additive relationships/smooth prediction surfaces.
  - Highly sensitive to changes in data (both with respect to observations and predictors).
  - Requires researchers to set “tuning’ ’ parameters (e.g. tree depth).
  - No natural measure of uncertainty.
- Ensembles of trees can help with many of these issues (for a price in terms of interpretability).

Tree ensembles: Bagging, boosting, and BART

$$f(X_i) = \sum_{m=1}^{M} \nu T_m(X_i; \Theta_m),$$

where

- M is the number of trees, ν is the contribution of each tree to the expansion, and Θ_m are the parameters that define tree T_m.
- The models we cover differ only in
  - the ways that trees are constructed, and
  - the ways that the trees are weighted.

Bagging and random forests

- Bagging:
  - Sobriquet for *B*ootstrap-*agg*regating.
  - As all sum-of-trees models, can capture both multiplicative *and* linear effects (when individual trees are built using splits on a single variable).
  - Meant to address the high-variance issues that plague single-tree models.
- Take M random samples (with replacement) and fit a “deep’ ’ tree (so as to reduce bias) using recursive binary splitting (along with regularizing strategy) on each sample.
- Then the bootstrapped predicted value for any given observation is simply the average over individual tree predictions:

$$\hat{f}_{bag}(X_i) = \frac{1}{M} \sum_{m=1}^{M} T_m(X_i; \hat{\Theta}_m)$$

Bagging and random forests

- Random forests (Breiman 2001):
    - Variance is only reduced if trees are uncorrelated.
    - Goal is to decrease dependence between individual tree predictions:
        - During recursive binary splitting, use only a fraction $a < 1$ of randomly selected covariates.
        - Use only subset of for
    - By only using a subset of the data, RF can perform a type of on-the-fly cross-validation.
- Usually better than CART, but not great at picking up on sparse regions of the covariate space.

Gradient boosting machines

- Like Bagging and RF, GBM (e.g. Freund and Schapire 1997) is a sum-of-trees model. Unlike them,
  - We add trees to the ensemble sequentially; ensemble is built in a forward stagewise fashion.
  - We don't fit trees to subsets of the data directly, but rather to (more informative) transformations of the data.
- For each new tree in the sequence, we optimize:

$$\hat{\Theta}_m = \arg\min_{\Theta_m} \sum_{i=1}^{N} L(y_i, f_{m-1}(X_i) + T_m(X_i; \Theta_m)) \tag{1}$$

  - Forces each new tree to focus on the errors of its predecessors.
- This can be approximated by

$$\hat{\Theta}_m = \arg\min_{\Theta_m} \sum_{i=1}^{N} (-g_{im} - T_m(X_i, \Theta_m))^2,$$

where $\mathbf{g}_m$ is the gradient of the loss function.

  - What is the negative gradient of squared error loss?

Gradient boosting machines: Regularization

- To avoid over-fitting the data:
  - Set B low (although this also determines the degree of assumed interactions in the model)
  - Set ν low (values of 0.1 and 0.01 are standard choices; can be thought of as rate of learning).
  - Cross-validate to find the number M of trees that minimizes approximate generalization error.
    - ⋆ You can also define optimal stoping rule
- GBM is:
  - A much more general estimation strategy. Boosting can be used with any model as a base learner (e.g. you can boost GAMs), but trees are the most common choice.
  - Wicked fast.
  - Very flexible (specially when trees are not too deep, and learning rate is low).
  - But it requires bootstrapping to get uncertainty estimates.

BART

- Bayesian Additive Regression Trees (Chipman et al. 2010) are very similar to boosted trees models, but they
  - Make explicit distributional assumptions about the conditional mean of the outcome.
  - Regularize the contribution of each tree using priors on Θ_m (i.e. the tree depth, splitting rules, and terminal-node values).

$$y_i = \sum_{m=1}^{M} T_m(X_i; \Theta_m) + \epsilon_i, \qquad \text{with } \epsilon_i \sim N(0, \sigma^2)$$

- Samples from a posterior of "forests' ' using a back-fitting MCMC algorithm.
  - Provides uncertainty estimates for Θ_m.
- Default priors seem to work very well (no cross validation is usually needed).
- But original implementation in `R` is clunky and estimation is slow.
  - Better alternative: `bartMachine`!

Opening the blackbox: Interpreting results

- Does the variable contribute to the model's explanatory power?

$$\mathcal{I}_j^{Improve} = \left(\frac{1}{M} \sum_{m=1}^{M} \sum_{k:\in K_{mj}} i_k^2 \right)^{0.5}$$

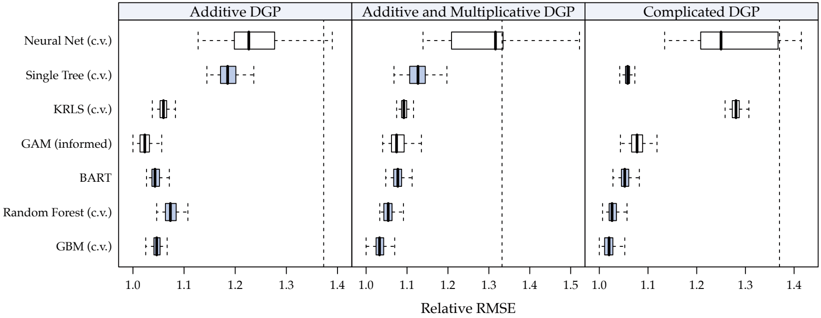
$$\mathcal{I}_j^{Use} = \frac{1}{S} \sum_{s=1}^{S} z_{js},$$

- Results on *bias*
  - Non-informative predictors should have an expected importance score of zero — they don't
    - ⋆ Correlated features, features with more splitting points
    - ⋆ Still,*relative* measures seem useful.

- What is the relationship between the covariate and the outcome?
  - Partial dependencies (known under different names in Political Science, including Average Predictive Effects):

$$\mathbb{E}[f(x_u, \mathbf{x}_{-u})] \doteq \bar{f}_u(x_u) = \frac{1}{N} \sum_{i=i}^{N} f(x_u, \mathbf{x}_{i,-u})$$

Comparing models



- Relative RMSE across 100 training sets for each model and each DGP specification. Lower values indicate better relative predictive accuracy with respect to test outcomes.

Applications I: Estimating sub-group preferences

- Multilevel regression and Post-stratification (MrP) models
  - Estimate quantities of interest at low levels of aggregation
    1. Model preferences using nationally representative surveys.
    2. Post-stratify group-level predictions using census frequencies.
- First step usually involves multi-level model:
  - Random intercepts by demographic groups and their intersections.
  - Random intercept by geographic unit.
  - Group level predictors.

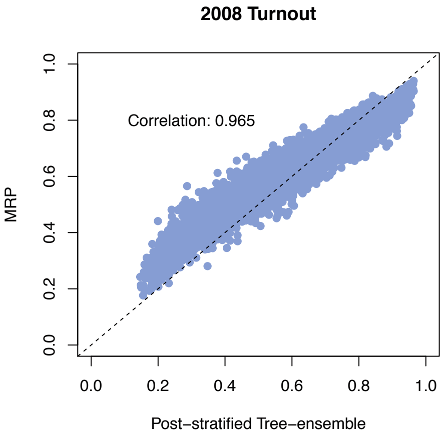
Applications I: Estimating sub-group preferences

- **Problem**: Fully-interactive multilevel models (MLM) face:
  - Computational limitations as the number of predictors increases
    - ★ Not just time, but feasibility
  - Issues with justification of functional form and covariate selection
    - ★ Theory may suggest many interactions, but not all are likely to matter.
    - ★ Why linear? Why Normal?
- **Solution**: Use tree-based ensembles to complete step 1.
  - Computationally efficient.
  - No need to specify functional form *a priori*.
    - ★ Automatic variable selection.
  - Flexible: can capture complicated relationships.

Applications I: Estimating sub-group preferences

- **Goal**: Obtain estimates of turnout intentions during the 2008 presidential election
    - For groups defined by ethnicity, income and age.
    - At state level
- **Data**:
    - Current Population Survey for turnout model (74,327 obs.).
    - American Community Survey for post-stratification.
- **Models**:
    - Fully-interactive, post-stratified binomial MLM (i.e. standard MrP for within-state subgroups): Ghitza and Gelman 2013.
    - Post-stratified tree-ensemble (viz. GBM) Montgomery & Olivella 2017.

Trees VS. MrP, Round 1: Speed



- Time to estimate:
  - MRP: About 1 hour and 20 minutes.
  - Trees model: **Less than 6 minutes**.

Trees VS. MrP, Round 2: Scalability

- **Previous subgroups**: state × ethnicity × income × age

Trees VS. MrP, Round 2: Scalability

- **Previous subgroups**: state × ethnicity × income × age
  - 4080 random intercepts.

Trees VS. MrP, Round 2: Scalability

- **Previous subgroups**: state × ethnicity × income × age
  - 4080 random intercepts.
- **Expanded subgroups**: state × ethnicity × income × age × sex × education × married × children

Trees VS. MrP, Round 2: Scalability

- **Previous subgroups**: state × ethnicity × income × age
    - 4080 random intercepts.
- **Expanded subgroups**:
  state × ethnicity × income × age × sex × education × married × children
    - 163200 random intercepts!

Trees VS. MrP, Round 2: Scalability

- **Previous subgroups**: state × ethnicity × income × age
  - 4080 random intercepts.
- **Expanded subgroups**:
  state × ethnicity × income × age × sex × education × married × children
  - 163200 random intercepts!
- Time to estimate:

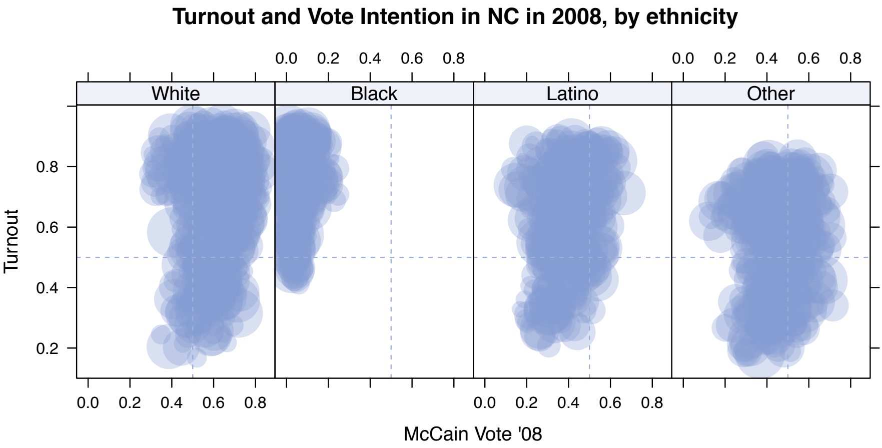
Trees VS. MrP, Round 2: Scalability

- **Previous subgroups**: state × ethnicity × income × age
  - 4080 random intercepts.
- **Expanded subgroups**:
  state × ethnicity × income × age × sex × education × married × children
  - 163200 random intercepts!
- Time to estimate:
  - MRP: ???

Trees VS. MrP, Round 2: Scalability

- **Previous subgroups**: state × ethnicity × income × age
    - 4080 random intercepts.
- **Expanded subgroups**:
  state × ethnicity × income × age × sex × education × married × children
    - 163200 random intercepts!
- Time to estimate:
    - MRP: ???
    - GBM: **about 8 minutes**.

Example: Racial and Partisan Gerrymandering in NC



Applications II: Propensity scores

- With observational data, it is still possible to identify average treatment effects if strong ignorability holds.
    - If it does, then conditioning/weighting on **true propensity score**, $e(\mathbf{x}) = \Pr(z = 1|\mathbf{x})$ alone is enough.
- But $e(\mathbf{x}) \neq \widehat{e(\mathbf{x})}$, and even small bias can be bad.
- **Problem**: Use of binomial regression to obtain $\widehat{e(\mathbf{x})}$ is a historical accident.
    - Even a very flexible binomial logit model is restrictive.
    - Open to researcher manipulation.
    - May not result in a balancing score.
- **Solution**: Use tree ensembles to estimate $e(\mathbf{x})$.
    - Very flexible, with minimal "researcher degrees of freedom".
    - Use justifiable methods to choose tuning parameters (e.g. CV).
    - Similar to subclassification using an (auto) coarsened covariate space.

Applications II: Propensity scores

- **Goal**: Obtain dynamic treatment effects of negative campaigning on candidate vote shares, in the presence of time-varying confounders, using IPTW.
- **Data**:
  - University of Wisconsin Advertising Project (1,150 obs.)
- **Models**: MSM model, with stabilized weights estimated using
  - Logit and carefully tailored GAM: Blackwell 2013 (Appendix: specification)
  - Cross-validated tree-ensembles (viz. GBM and BART): Montgomery & Olivella 2017

GAM IPTW VS. Tree IPTW

	Non-incumbents	Incumbents
Unweighted estimate	0.450	−0.909
GAM weights	0.71	−0.553
GBM weights	0.544	−0.630
BART weights	0.521	−0.563

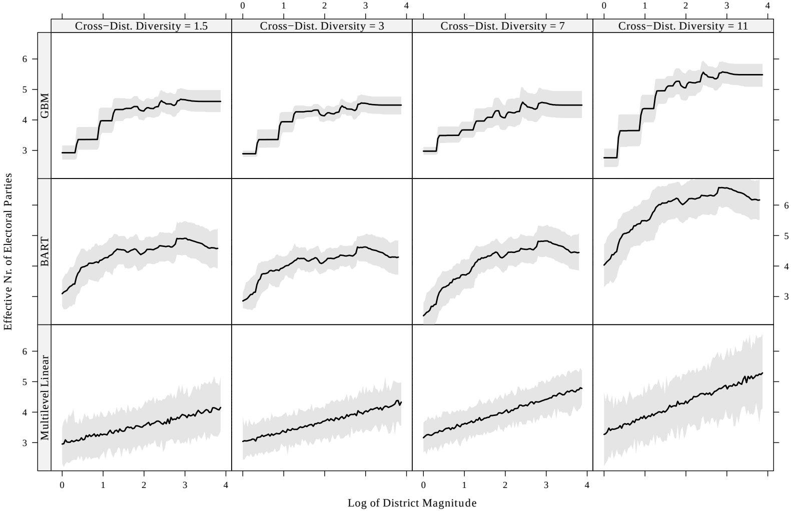
- Tree weighted MSM is more consistent with existent literature; MSM with GAM is not.
- No researcher-driven specification search was involved.

Applications III: Duverger's Law

- Importance indicators for covariates determining district-level effective number of parties

	$I_j^{Improve}$	I_j^{Use}
Model	GBM	BART
District magnitude	1.00	0.23
Cross-district diversity	0.453	0.21
Age of democratic system	0.291	0.23
District diversity	0.033	0.16
Mixed system	0.019	0.16
Out-of-sample RMSE	0.67	0.69
Out-of-sample R^2	0.57	0.55
n	1581	1581

Duverger's Law: Partial dependence plots



A note on trees for theory testing

- The issue is similar to that raised by Breiman (2001) in his “Two cultures”:
  - **Algorithmic modeling**: Try to reproduce nature’s blackbox in all its complexity, and use artificial “nature” to get predictions and check expectations.
- Also similar to points raised by King and Nielsen (2016)
  - Historically, we’ve used statistics to reduce arbitrary researcher input in data analysis (think participant observation). Using blackbox methods is a natural step forward.

Concluding remarks: Trees for theory testing

- Tree ensembles are good options when goal is unbiased estimation of intermediate quantity.
  - Exact functional form unimportant, provided it is correct.
- What about using trees for evaluating evidence on $\frac{\partial y}{\partial x}$?
  - Often, theories produce simple expectations about effects:
    - ⋆ Direction
    - ⋆ Conditionality
    - ⋆ Convexity
- We can always get arbitrarily good approximations of $\frac{\partial y}{\partial x}$ using predicted values.
- "But models should be chosen on theoretical grounds!"
  - Partly. A model should reflect relevant aspects of reality, so we can evaluate consequences of controlled "tweaks".
  - Without comparison to alternative models, significant results are merely constructive proofs
- Nevertheless:
  - Parametric theories require parametric models
  - There is no substitute for good design: no model can solve selection on