Tree-based models for political science data

November 30, 2017

Flexible modeling of data

- Handling sparcity

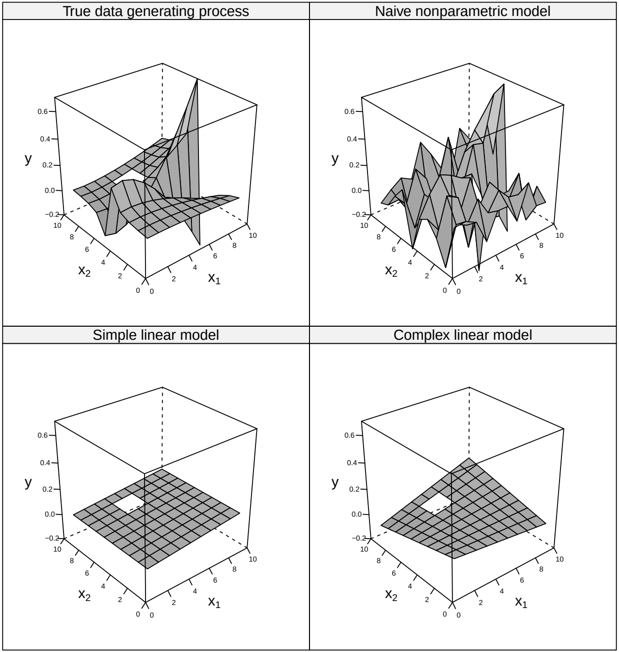
Flexible modeling of data

- Handling sparcity
- Accuracy

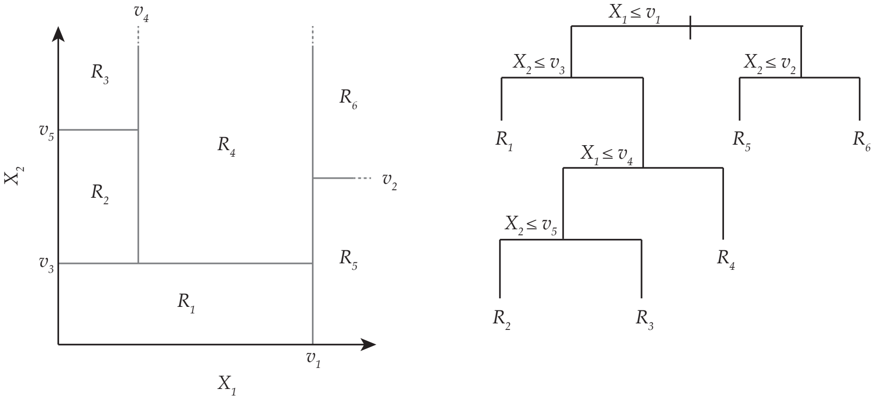
Flexible modeling of data

- Handling sparcity
- Accuracy
- Regularization

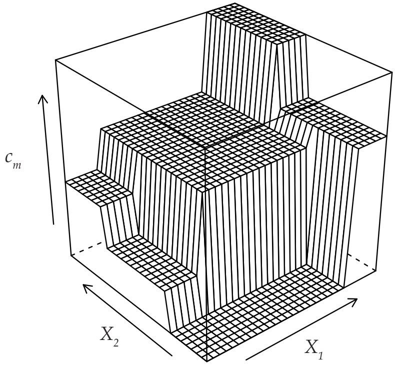
X_1 \ X_2	1	2	3	4	5	6	7	8	9	10
10	4	9	6	9	4	5	6	5	5	3
9	7	8	4	7	4	13	12	5	4	3
8	5	3	7	7		8	6	8	3	6
7	2	4	6	6	5	13	12	4	5	5
6	7	5	6	8	4	7	4	6	4	5
5	2	6	11	11	2	11	4	2	3	6
4	5	6	6	5	3	9	5	4	3	5
3	7	6	11	5	5	4	2	4	8	5
2	4	7	8	7	10	7	9	8	2	7
1	8	10	5	7	5	5	13	2	9	7


True data generating process
Naive nonparametric model
Simple linear model
Complex linear model
y
x_1
x_2

Partitioning the covariate space using binary trees



Partitioning the covariate space using binary trees



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

The basics of single-tree models: Optimization

$$f(X_i) = T(X_i; \Theta) \equiv \sum_{b=1}^{B} c_b I(X_i \in R_b) \tag{2}$$

$$\hat{\Theta} = \arg\min_{\Theta} \sum_{b=1}^{B} \sum_{X_i \in R_b} L(y_i, c_b) \tag{3}$$

Where $L(.)$ is some loss function, e.g., $\sum_{i:X_i \in R_b} (y_i - c_b)^2$.

The basics of single-tree models: Regularization

	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.

The basics of single-tree models: 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 $C_\alpha(T) = \sum_{b=1}^{B} L(y_{i:X_i \in R_b}, c_b) + \alpha B$
- B is the number of terminal nodes
- $\alpha \geq 0$ is user specified

Pros and cons of single-tree models

Pros:

- Intuitive

Pros and cons of single-tree models

Pros:

- Intuitive
- Handles lot's of data issues smoothly

Pros and cons of single-tree models

Pros:

- Intuitive
- Handles lot's of data issues smoothly
- Easily handles interactions

Pros and cons of single-tree models

Pros:

- Intuitive
- Handles lot's of data issues smoothly
- Easily handles interactions

Pros and cons of single-tree models

Pros:

- Intuitive
- Handles lot's of data issues smoothly
- Easily handles interactions

Cons:

- Performs poorly for additive relationships

Pros and cons of single-tree models

Pros:

- Intuitive
- Handles lot's of data issues smoothly
- Easily handles interactions

Cons:

- Performs poorly for additive relationships
- Highly sensitive to subsetting

Pros and cons of single-tree models

Pros:

- Intuitive
- Handles lot's of data issues smoothly
- Easily handles interactions

Cons:

- Performs poorly for additive relationships
- Highly sensitive to subsetting
- No uncertainty

Pros and cons of single-tree models

Pros:

- Intuitive
- Handles lot's of data issues smoothly
- Easily handles interactions

Cons:

- Performs poorly for additive relationships
- Highly sensitive to subsetting
- No uncertainty

Pros and cons of single-tree models

Pros:

- Intuitive
- Handles lot's of data issues smoothly
- Easily handles interactions

Cons:

- Performs poorly for additive relationships
- Highly sensitive to subsetting
- No uncertainty

Solution is to use ensembles of trees

Tree ensembles: Bagging, boosting, and BART

$$f(X_i) = \sum_{m=1}^{M} T_m(X_i; \Theta_m), \tag{4}$$

- M is the number of trees, and Θ_m are the parameters that define tree T_m

Tree ensembles: Bagging, boosting, and BART

$$f(X_i) = \sum_{m=1}^{M} T_m(X_i; \Theta_m), \tag{4}$$

- M is the number of trees, and Θ_m are the parameters that define tree T_m
- The models we cover differ only in

Tree ensembles: Bagging, boosting, and BART

$$f(X_i) = \sum_{m=1}^{M} T_m(X_i; \Theta_m), \tag{4}$$

- M is the number of trees, 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

Tree ensembles: Bagging, boosting, and BART

$$f(X_i) = \sum_{m=1}^{M} T_m(X_i; \Theta_m), \tag{4}$$

- M is the number of trees, 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.

Tree ensembles: Bagging, boosting, and BART

$$f(X_i) = \sum_{m=1}^{M} T_m(X_i; \Theta_m), \tag{5}$$

- M is the number of trees, and Θ_m are the parameters that define tree T_m

Tree ensembles: Bagging, boosting, and BART

$$f(X_i) = \sum_{m=1}^{M} T_m(X_i; \Theta_m), \tag{5}$$

- M is the number of trees, and Θ_m are the parameters that define tree T_m
- The models we cover differ only in

Tree ensembles: Bagging, boosting, and BART

$$f(X_i) = \sum_{m=1}^{M} T_m(X_i; \Theta_m), \tag{5}$$

- M is the number of trees, 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

Tree ensembles: Bagging, boosting, and BART

$$f(X_i) = \sum_{m=1}^{M} T_m(X_i; \Theta_m), \quad (5)$$

- M is the number of trees, 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:

- Bootstrap-aggregation

Bagging and random forests

Bagging:

- Bootstrap-aggregation
- Take random samples (with replacement) and fit a “deep” tree

Bagging and random forests

Bagging:

- Bootstrap-aggregation
- Take random samples (with replacement) and fit a “deep” tree
- $\hat{f}_{bag}(X_i) = \frac{1}{M}\sum_{m=1}^{M} T_m(X_i; \hat{\Theta}_m)$

Bagging and random forests

Bagging:

- Bootstrap-aggregation
- Take random samples (with replacement) and fit a "deep" tree
- $\hat{f}_{bag}(X_i) = \frac{1}{M} \sum_{m=1}^{M} T_m(X_i; \hat{\Theta}_m)$

Bagging and random forests

Bagging:

- Bootstrap-aggregation
- Take random samples (with replacement) and fit a “deep” tree
- $\hat{f}_{bag}(X_i) = \frac{1}{M}\sum_{m=1}^{M} T_m(X_i; \hat{\Theta}_m)$

Random forests:

- Goal is to decrease dependence between samples

Bagging and random forests

Bagging:

- Bootstrap-aggregation
- Take random samples (with replacement) and fit a “deep” tree
- $\hat{f}_{bag}(X_i) = \frac{1}{M}\sum_{m=1}^{M} T_m(X_i; \hat{\Theta}_m)$

Random forests:

- Goal is to decrease dependence between samples
- During recursive binary splitting, use only $a < j$ randomly selected covariates.

Gradient boosting machines: Optimization

Unlike bagging:

- We add trees to the ensemble sequentially

Gradient boosting machines: Optimization

Unlike bagging:

- We add trees to the ensemble sequentially
- We don't fit trees to subsets of the data but rather transformations of the data.

Gradient boosting machines: Optimization

Unlike bagging:

- We add trees to the ensemble sequentially
- We don't fit trees to subsets of the data but rather transformations of the data.

Gradient boosting machines: Optimization

Unlike bagging:

- We add trees to the ensemble sequentially
- We don't fit trees to subsets of the data but rather transformations of the data.

For each new tree, we now maximize:

$$\hat{\Theta}_m = \arg\min_{\Theta_m} \sum_{i=1}^{N} L\big(y_i, f_{m-1}(X_i) + T_m(X_i; \Theta_m)\big) \tag{6}$$

Since this cannot be calculated directly, we approximate using:

$$\hat{\Theta}_m = \arg\min_{\Theta_m} \sum_{i=1}^{N} \big(-g_{im} - T_m(X_i, \Theta_m)\big)^2, \tag{7}$$

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

Gradient boosting machines: Regularization

To avoid over-fitting the data:

- Set B very low (although this also determines the degree of assumed interactions in the model)
- Shrinkage: $f(X_i) = f_{m-1}(X_i) + \nu T(X_i; \Theta_m)$

Gradient boosting machines: Regularization

To avoid over-fitting the data:

- Set B very low (although this also determines the degree of assumed interactions in the model)
- Shrinkage: $f(X_i) = f_{m-1}(X_i) + \nu T(X_i; \Theta_m)$

GBM is:

- wicked fast
- very flexible
- requires bootstrapping to get uncertainty estimates

BART

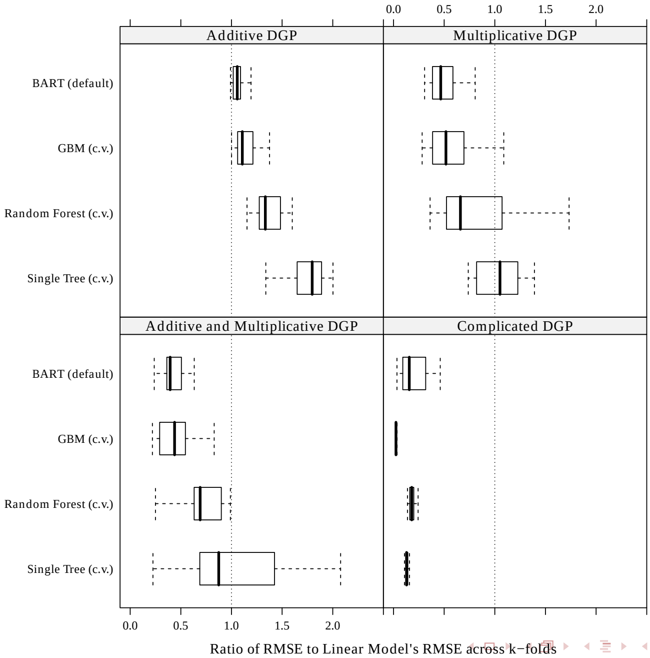
$$y_i = \sum_{m=1}^{M} T_m(X_i; \Theta_m) + \epsilon_i, \qquad \text{with } \epsilon_i \sim N(0, \sigma^2) \tag{8}$$

BART

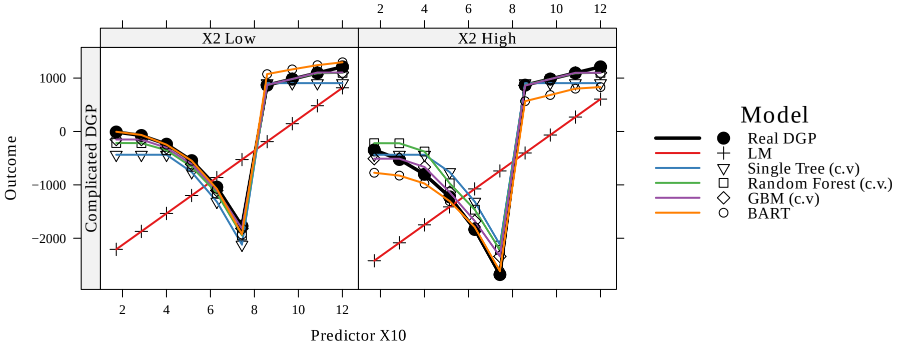
$$y_i = \sum_{m=1}^{M} T_m(X_i; \Theta_m) + \epsilon_i, \qquad \text{with } \epsilon_i \sim N(0, \sigma^2) \tag{8}$$

- Samples from a posterior of ensemble models
- Provides uncertainty estimates
- Default priors seem to work very well (no cross validation)
- Not as flexible and SLOW

Model	Tuning parameters	Usual Values
Single tree (`rpart`)	Min. nr. of observations in any terminal node (`minbucket`)	3, 5, 10
	Max. tree depth (`maxdepth`)	3, 5, 15
	Minimum factor decrease in lack-of-fit measure (`cp`)	Inf., 0.1, 0.01, 0.001
Random forest (`randomForest`)	Nr. of trees (`ntree`)	100, 500, 1000
	Nr. of candidate variables sampled at each split (`mtry`)	2, 10
	Min. nr. of observations in any terminal node (`nodesize`)	3, 5, 10
	Nr. of observations sampled when forming each tree (`sampsize`)	360, 720
Gradient boosting (`gbm`)	Nr. of trees (`n.trees`)	100, 500, 1000
	Max. tree depth (`interaction.depth`)	3, 5, 10
	Learning rate (or factor shrinkage) of each tree (`shrinkage`)	0.001, 0.005, 0.01
BART (`BayesTree`)	Degrees of freedom for error variance prior (`sigdf`)	3, 10
	Quantile for error variance prior (`sigquant`)	0.9, 0.99, 0.75
	Nr. of trees (`ntree`)	50, 200
	Factor multiplying the prior range of the outcome variable (`k`)	1, 2, 3, 4


Additive DGP
Multiplicative DGP
Additive and Multiplicative DGP
Complicated DGP
BART (default)
GBM (c.v.)
Random Forest (c.v.)
Single Tree (c.v.)
0.0
0.5
1.0
1.5
2.0
Ratio of RMSE to Linear Model's RMSE across k−folds

Recovering relationships: Synthetic data



Predicted values of outcome variable as a function of x_{10} under the complicated DGP for tree-based models and a linear model, conditional on two values of x_2. The thicker, black line with solid circles represents the true effect of x_{10}.

Example: Evaluating Duverger's Law

- District magnitude $\rightarrow$ number of parties

Example: Evaluating Duverger's Law

- District magnitude $\rightarrow$ number of parties
- This relationship is:

Example: Evaluating Duverger's Law

- District magnitude $\rightarrow$ number of parties
- This relationship is:
  - Non-linear

Example: Evaluating Duverger's Law

- District magnitude $\rightarrow$ number of parties
- This relationship is:
  - Non-linear
  - Interactive with social diversity (e.g, Clark and Golder)

Example: Evaluating Duverger's Law

- District magnitude $\rightarrow$ number of parties
- This relationship is:
  - Non-linear
  - Interactive with social diversity (e.g, Clark and Golder)
- We use data from Potter (2014), which has 1500 election-district observations where we have:

Example: Evaluating Duverger's Law

- District magnitude $\rightarrow$ number of parties
- This relationship is:
  - Non-linear
  - Interactive with social diversity (e.g, Clark and Golder)
- We use data from Potter (2014), which has 1500 election-district observations where we have:
  - effective number of parties

Example: Evaluating Duverger's Law

- District magnitude $\rightarrow$ number of parties
- This relationship is:
  - Non-linear
  - Interactive with social diversity (e.g, Clark and Golder)
- We use data from Potter (2014), which has 1500 election-district observations where we have:
  - effective number of parties
  - log of district magnitude

Example: Evaluating Duverger's Law

- District magnitude $\rightarrow$ number of parties
- This relationship is:
  - Non-linear
  - Interactive with social diversity (e.g, Clark and Golder)
- We use data from Potter (2014), which has 1500 election-district observations where we have:
  - effective number of parties
  - log of district magnitude
  - demographic groups within district

Example: Evaluating Duverger's Law

- District magnitude $\rightarrow$ number of parties
- This relationship is:
  - Non-linear
  - Interactive with social diversity (e.g, Clark and Golder)
- We use data from Potter (2014), which has 1500 election-district observations where we have:
  - effective number of parties
  - log of district magnitude
  - demographic groups within district
  - demographic groups at the national level

Example: Evaluating Duverger's Law

- District magnitude $\rightarrow$ number of parties
- This relationship is:
  - Non-linear
  - Interactive with social diversity (e.g, Clark and Golder)
- We use data from Potter (2014), which has 1500 election-district observations where we have:
  - effective number of parties
  - log of district magnitude
  - demographic groups within district
  - demographic groups at the national level
  - dummy for mixed systems

Example: Evaluating Duverger's Law

- District magnitude $\rightarrow$ number of parties
- This relationship is:
  - Non-linear
  - Interactive with social diversity (e.g, Clark and Golder)
- We use data from Potter (2014), which has 1500 election-district observations where we have:
  - effective number of parties
  - log of district magnitude
  - demographic groups within district
  - demographic groups at the national level
  - dummy for mixed systems
  - age of the democratic system

Example: Evaluating Duverger's Law

- District magnitude $\rightarrow$ number of parties
- This relationship is:
  - Non-linear
  - Interactive with social diversity (e.g, Clark and Golder)
- We use data from Potter (2014), which has 1500 election-district observations where we have:
  - effective number of parties
  - log of district magnitude
  - demographic groups within district
  - demographic groups at the national level
  - dummy for mixed systems
  - age of the democratic system
- We fit the model using GBM (cv) and BART (default)

Interpreting results

1. Does the variable contribute to the model's explanatory power? (i.e., How much does this variable improve the fit of the overall model?)
2. What is the relationship between the covariate and the outcome? (i.e., Does the conditional relationship between the variable and the outcome of interest match theoretical expectations?)

Which variables are important?

$$\mathcal{I}_j^{Improve} = \left(\frac{1}{M} \sum_{m=1}^{M} \sum_{k:\in K_{mj}} i_k^2 \right)^{0.5} \tag{9}$$

Which variables are important?

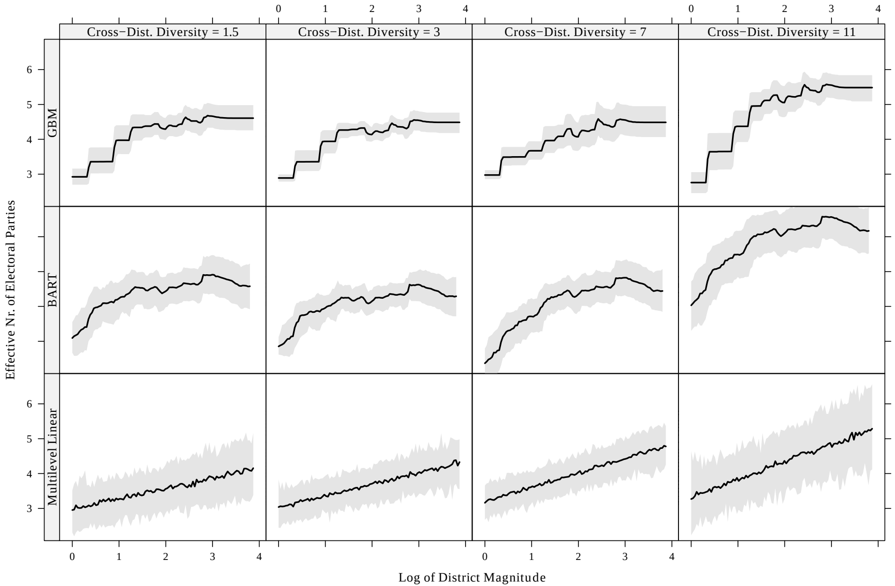
$$\mathcal{I}_j^{Improve} = \left(\frac{1}{M} \sum_{m=1}^{M} \sum_{k:\in K_{mj}} i_k^2 \right)^{0.5} \tag{9}$$

$$\mathcal{I}_j^{Use} = \frac{1}{S} \sum_{s=1}^{S} z_{js}, \tag{10}$$

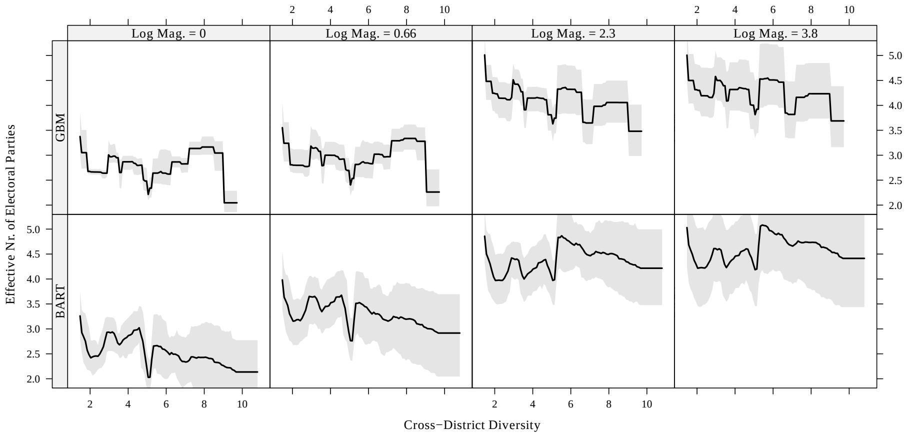
Importance indicators for covariates determining district-level effective number of parties

	$I_j^{Improve}$	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

Partial dependence plots



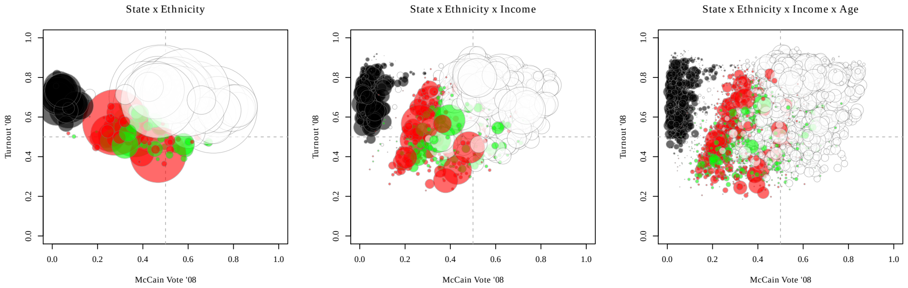
Partial dependence plots: Removing Brazil/Italy



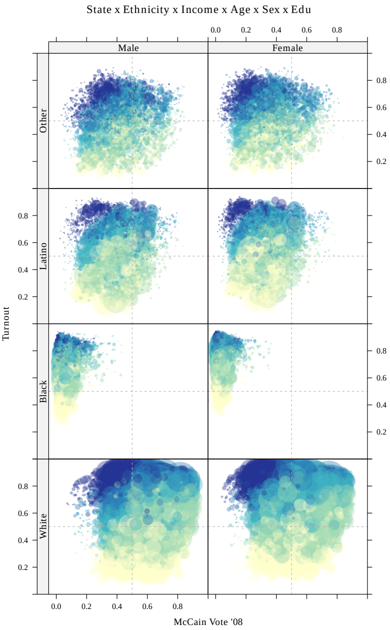
Example: Estimating opinion of demographic sub-groups

- Ghitza and Gelman (2013) use MRP with high-level interactions to model demographic sub-groups
- "By modeling deeper levels of interactions and allowing for the relationship between covariates to be non-linear and even non-monotonic" (p. 773).
- The method does not allow for all levels of interactions.
- We are going to use BART-P to discover even deeper patterns in the data.
- Data are from the 2004 (n=43,970) and 2008 (n=19,170) NAES

Replicating Ghitza and Gelman using BART



- R=0.974 for 2008 McCain Vote
- R=0.98 for 2008 Turnout


State x Ethnicity x Income x Age x Sex x Edu
Male
Female
Other
Latino
Black
White
Turnout
McCain Vote '08
0.0
0.2
0.4
0.6
0.8

Summary

The good:

- Valuable tool for non-parametric modeling of large datasets
- Particularly valuable for detecting non-linearities, interactions, and many covariates
- Given the direction of our data resources, these methods should become a more standard tool in the discipline.

Summary

The good:

- Valuable tool for non-parametric modeling of large datasets
- Particularly valuable for detecting non-linearities, interactions, and many covariates
- Given the direction of our data resources, these methods should become a more standard tool in the discipline.

The bad:

- Does not provide either theory or identification
- Don't be evil.

Outline

- The missing data problem
- Types of missingness
- Multiple imputation

Missing Data

- What is missing data?
  - It is when we have no data value for a variable in an observation.
  - e.g., Surveys in which some respondents did not answer certain questions.

Missing Data

- What is missing data?
  - It is when we have no data value for a variable in an observation.
  - e.g., Surveys in which some respondents did not answer certain questions.

- What is **not** missing data?
  - Missing all values of a variable for certain "groups"
  - e.g., A survey question was not asked in a particular country, a survey respondent that did not answer any question
  - This is related to *case selection*.

Missing Data

- What is missing data?
  - It is when we have no data value for a variable in an observation.
  - e.g., Surveys in which some respondents did not answer certain questions.
- What is **not** missing data?
  - Missing all values of a variable for certain "groups"
  - e.g., A survey question was not asked in a particular country, a survey respondent that did not answer any question
  - This is related to *case selection*.
- What is the most common method of addressing missing data?
  - Casewise deletion or list-wise deletion or complete case analysis

Mechanisms by which Data can be Missing

R is a matrix that with a dichotomous indicator valued 1 if a datum in **X** is missing and 0 if it is not. The missing data generating mechanism is described by ϕ (Little and Rubin, 2002, p.12)

$$Z_{mis} = (X_{mis}, Y_{mis})$$
$$Z_{obs} = (X_{obs}, Y_{obs})$$

Mechanisms by which Data can be Missing

R is a matrix that with a dichotomous indicator valued 1 if a datum in **X** is missing and 0 if it is not. The missing data generating mechanism is described by ϕ (Little and Rubin, 2002, p.12)

$$Z_{mis} = (X_{mis}, Y_{mis})$$
$$Z_{obs} = (X_{obs}, Y_{obs})$$

- Missing completely at random (MCAR)
  - Missingness not related to any observed or unobserved data

$$P(\mathtt{R}|Z_{obs}, Z_{mis}) = P(R|\phi)$$

Mechanisms by which Data can be Missing

R is a matrix that with a dichotomous indicator valued 1 if a datum in **X** is missing and 0 if it is not. The missing data generating mechanism is described by ϕ (Little and Rubin, 2002, p.12)

$$Z_{mis} = (X_{mis}, Y_{mis})$$
$$Z_{obs} = (X_{obs}, Y_{obs})$$

- Missing completely at random (MCAR)
  - Missingness not related to any observed or unobserved data

$$P(\mathtt{R}|Z_{obs}, Z_{mis}) = P(R|\phi)$$

- Missing at random (MAR)
  - Missingness depends only on observed data

$$P(\mathtt{R}|Z_{obs}, Z_{mis}) = P(\mathtt{R}|Z_{obs}, \phi)$$

Mechanisms by which Data can be Missing

R is a matrix that with a dichotomous indicator valued 1 if a datum in **X** is missing and 0 if it is not. The missing data generating mechanism is described by ϕ (Little and Rubin, 2002, p.12)

$$Z_{mis} = (X_{mis}, Y_{mis})$$
$$Z_{obs} = (X_{obs}, Y_{obs})$$

- Missing completely at random (MCAR)
  - Missingness not related to any observed or unobserved data
$$P(\mathtt{R}|Z_{obs}, Z_{mis}) = P(R|\phi)$$
- Missing at random (MAR)
  - Missingness depends only on observed data
$$P(\mathtt{R}|Z_{obs}, Z_{mis}) = P(\mathtt{R}|Z_{obs}, \phi)$$
- Non-ignorable (NI)
  - Missingness depends on unobserved data
$$P(\mathtt{R}|Z_{obs}, Z_{mis}) = P(\mathtt{R}|Z_{obs}, Z_{mis}, \phi)$$

Examples

- Missing Completely at Random (MCAR)
  - Respondents accidentally skip questions.

Examples

- Missing Completely at Random (MCAR)
    - Respondents accidentally skip questions. Very unlikely.
- Missing at Random (MAR)
    - Respondents with low political sophistication, low levels of information and education, do not know how to place themselves of the liberal/conservative dimension.
    - Respondents with lower income do not answer questions about their income.

Examples

- Missing Completely at Random (MCAR)
  - Respondents accidentally skip questions. Very unlikely.
- Missing at Random (MAR)
  - Respondents with low political sophistication, low levels of information and education, do not know how to place themselves of the liberal/conservative dimension.
  - Respondents with lower income do not answer questions about their income.
- Non-Ignorable
  - Randomized experiment about voter turnout with a pre-election survey and a post-election survey. Non-response rate to the post-election survey is lower for the treatment group than for the control group. The missing data mechanism depends on the actual turnout of the participants of the experiment, which we don't have!

Why Case-wise Deletion is Evil

- Consider the computation of a mean μ from data $\mathbf{y}$ where some data are non-randomly missing.
- When μ_R is the mean of respondents and μ_M is the mean of missing data, we write the overall mean as:

$$\mu = \pi_R \mu_R + (1 - \pi_R)\mu_M$$

where π_R is the *proportion* of observed responses.

Why Case-wise Deletion is Evil

- Consider the computation of a mean μ from data $\mathbf{y}$ where some data are non-randomly missing.
- When μ_R is the mean of respondents and μ_M is the mean of missing data, we write the overall mean as:

$$\mu = \pi_R \mu_R + (1 - \pi_R)\mu_M$$

  where π_R is the *proportion* of observed responses.
- The bias produced by casewise deletion is the expected fraction of missing data times the difference in means for observed and missing data (Little and Rubin, 2002, p.43):

$$\mu_R - \mu = (1 - \pi_R)(\mu_R - \mu_M)$$

- In the special case MCAR, $\mu_R = \mu_M$ and the statistic is unbiased, but this is commonly violated in the social sciences.

Multiple Imputation (Rubin 1979). Steps.

Steps:

1. generate reasonable values for the missing the data (Z_{mis}) m times to get m replicate datasets,

Multiple Imputation (Rubin 1979). Steps.

Steps:

1. generate reasonable values for the missing the data (Z_{mis}) m times to get m replicate datasets,
2. analyze/regress each dataset separately,

Multiple Imputation (Rubin 1979). Steps.

Steps:

1. generate reasonable values for the missing the data (Z_{mis}) m times to get m replicate datasets,
2. analyze/regress each dataset separately,
3. combine results with summary process.

Multiple Imputation (Rubin 1979). Steps.

Steps:

1. generate reasonable values for the missing the data (Z_{mis}) m times to get m replicate datasets,
2. analyze/regress each dataset separately,
3. combine results with summary process.

- Imputation step assumes missing data is conditioned on observed values.
- Oddly, enough m = 5 to 10 is sufficient.
- Combining process uses means for coefficients and an intuitive approach for standard errors.

Rubin's rule

$$Var_{within} = \frac{\sum_{i=1}^{M} SE_i^2}{M}$$

$$Var_{between} = \frac{\sum_{i=1}^{M} (\beta_i - \bar{\beta})^2}{M-1}$$

$$Var_{total} = Var_{within} + Var_{between} + \frac{Var_{between}}{M}$$

Now in vector format

$$\bar{\beta} = \frac{\sum_{i=1}^{n} \hat{\beta}_i}{n}$$

Now in vector format

$$\bar{\boldsymbol{\beta}} = \frac{\sum_{i=1}^{n} \hat{\boldsymbol{\beta}}_i}{n}$$

$$\boldsymbol{\Omega} = \frac{1}{m} \sum_{i=1}^{m} \hat{\mathbf{V}}_i$$

where $\hat{\mathbf{V}}_i$ is the variance covariance matrix for $\hat{\boldsymbol{\beta}}_i$

Now in vector format

$$\bar{\beta} = \frac{\sum_{i=1}^{n} \hat{\beta}_i}{n}$$

$$\mathbf{\Omega} = \frac{1}{m} \sum_{i=1}^{m} \hat{\mathbf{V}}_i$$

where $\hat{\mathbf{V}}_i$ is the variance covariance matrix for $\hat{\beta}_i$

$$\mathbf{\Psi} = \frac{1}{m-1} \sum_{i=1}^{m} (\hat{\beta}_i - \bar{\beta})'(\hat{\beta}_i - \bar{\beta})$$

Now in vector format

$$\bar{\beta} = \frac{\sum_{i=1}^{n} \hat{\beta}_i}{n}$$

$$\mathbf{\Omega} = \frac{1}{m} \sum_{i=1}^{m} \hat{\mathbf{V}}_i$$

where $\hat{\mathbf{V}}_i$ is the variance covariance matrix for $\hat{\beta}_i$

$$\mathbf{\Psi} = \frac{1}{m-1} \sum_{i=1}^{m} (\hat{\beta}_i - \bar{\beta})'(\hat{\beta}_i - \bar{\beta})$$

$$\text{Total Variance} = \mathbf{\Omega} + \mathbf{\Psi} + \mathbf{\Omega}/m$$

Practical Issues

- Pre-specify if a variable is a factor, numeric, etc. Read manuals.

Practical Issues

- Pre-specify if a variable is a factor, numeric, etc. Read manuals.
- Dataset cannot have perfectly collinear variables. e.g, A variable with country names and another variable with country IDs.

Practical Issues

- Pre-specify if a variable is a factor, numeric, etc. Read manuals.
- Dataset cannot have perfectly collinear variables. e.g, A variable with country names and another variable with country IDs.
- Dataset for imputation usually has more variables than the model we want to specify.

Practical Issues

- Pre-specify if a variable is a factor, numeric, etc. Read manuals.
- Dataset cannot have perfectly collinear variables. e.g, A variable with country names and another variable with country IDs.
- Dataset for imputation usually has more variables than the model we want to specify.
- "An imputation model does not represent causal relationships among the data." (Young and Johnson 2010)

Missing in `R`

- `mice` (van Buuren et al. 2006, van Buuren 2007)
- `Amelia`, easy to use (King, Honaker, Blackwell)
- `mi`, for multilevel data (Kropko, Gelman, others)
- `hot.deck`, for categorical variables (Cranmer and Gill 2013)