Classification and Regression Trees (CART) and Causal Trees

Susan Athey, Stanford University

Machine Learning and Causal Inference

What is the goal of prediction?

- Machine learning answer:
  - Smallest mean-squared error in a test set

- Formally:
  - Let S^{te} be a test set.
    - Think of this as a random draw of individuals from a population
  - Let $\hat{\mu}(x_i)$ be a candidate (estimated) predictor
  - MSE on test set is:

$$\frac{1}{|S^{te}|}\sum_{i \in S^{te}} (Y_i - \hat{\mu}(X_i))^2$$

Regression Trees

- Simple method for prediction
  - Partition data into subsets by covariates
  - Predict using average within each subset

- Why are regression trees popular?
  - Easy to understand and explain
  - Businesses often need "segments"
  - Software assigns different algorithms to different segments

- Can completely describe the algorithm and interpretation

Example: Who survived the Titantic?

Regression Trees for Prediction

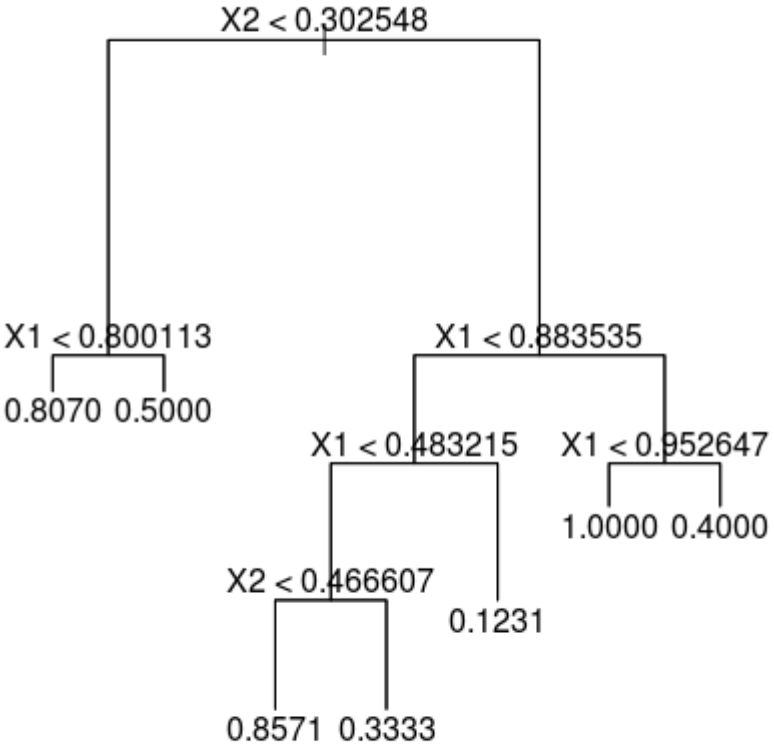
Data

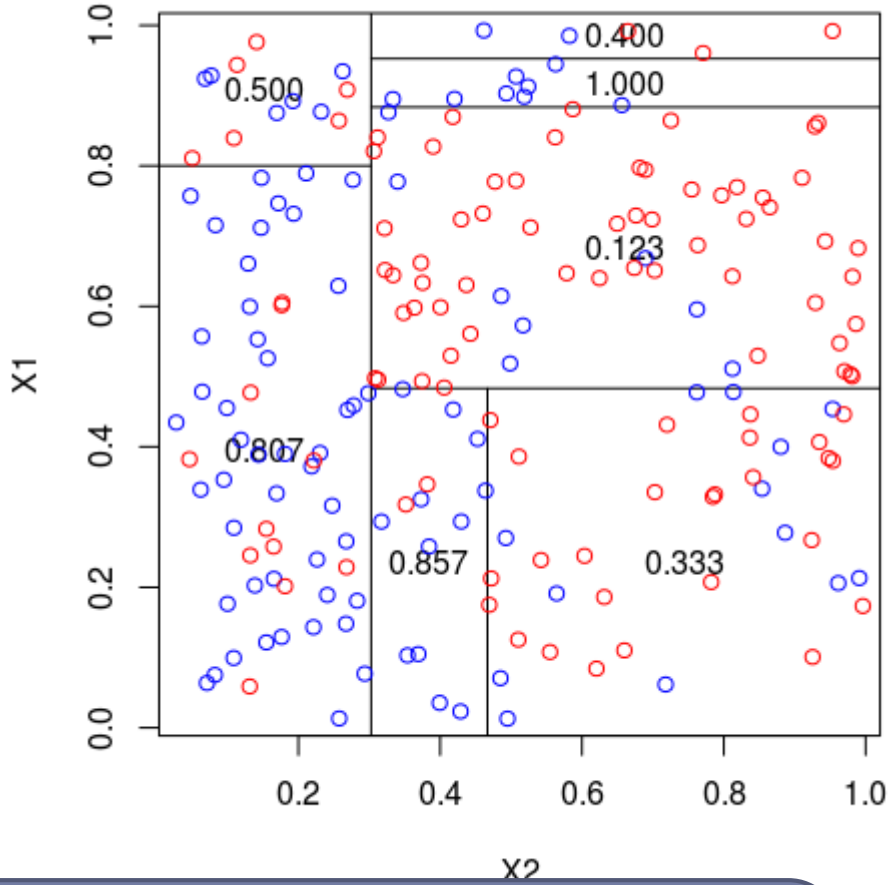
- Outcomes Y_i, attributes X_i.
- Support of X_i is $\mathcal{X}$.
- Have training sample with independent obs.
- Want to predict on new sample

Build a "tree":

- Partition of $\mathcal{X}$ into "leaves" $\mathcal{X}_j$
- Predict Y conditional on realization of X in each region $\mathcal{X}_j$ using the sample mean in that region
- Go through variables and leaves and decide whether and where to split leaves (creating a finer partition) using in-sample goodness of fit criterion
- Select tree complexity using cross-validation based on prediction quality

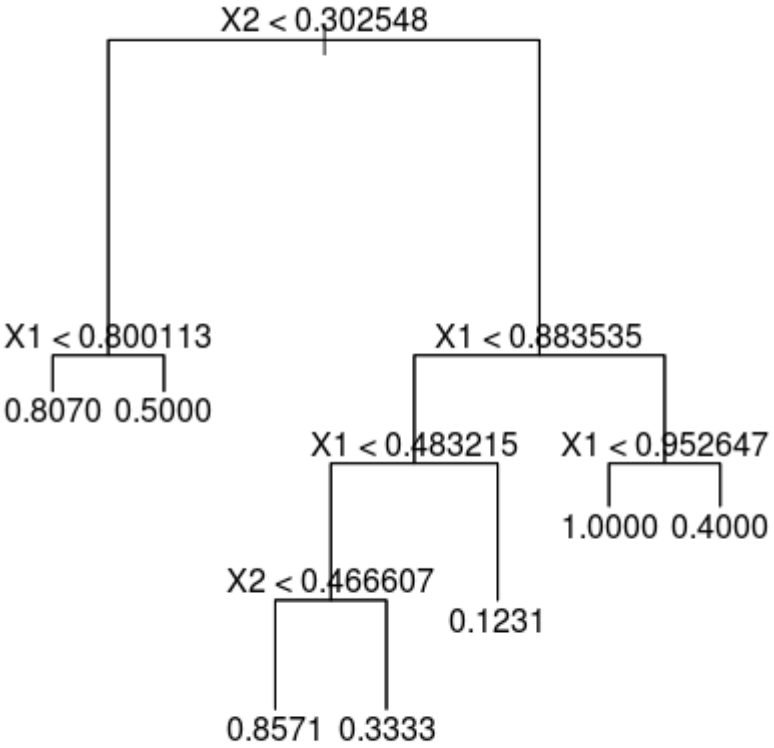
Regression Trees for Prediction

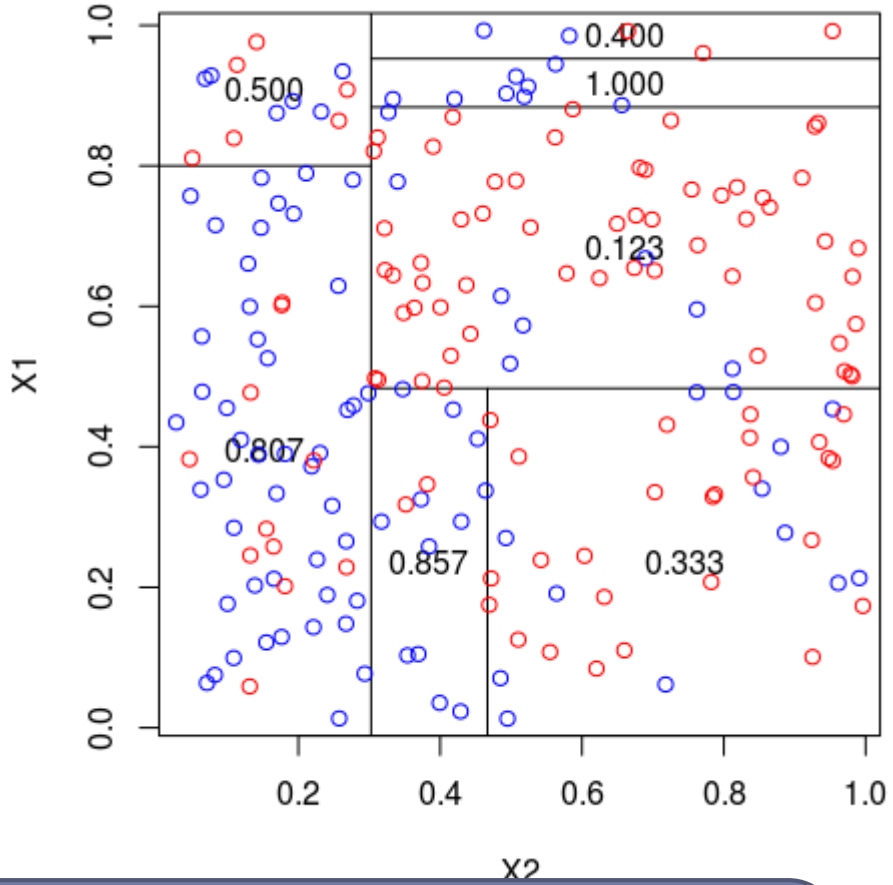




Outcome: Binary (Y in {0,1})
Two covariates
Goal: Predict Y as a function of X
"Classify" units as a function of X according to whether they are more likely to have Y=0 or Y=1

Regression Trees for Prediction





(I) Tree-building: Use algorithm to partition data according to covariates (adaptive: do this based on the difference in mean outcomes in different potential leaves.)
(II) Estimation/prediction: calculate mean outcomes in each leaf
(III) Use cross-validation to select tree complexity penalty

Tree Building Details

- Impossible to search over all possible partitions, so use a greedy algorithm
- Do until all leaves have less than 2*minsize obs:
  - For each leaf:
    - For each observed value $\tilde{x}_j$ of each covariate x_j:
      - Consider splitting the leaf into two children according to whether $\tilde{x}_j \leq x_j$
      - Make new predictions in each candidate child according to sample mean
      - Calculate the improvement in "fit" (MSE)
    - Select the covariate *j* and the cutoff value that lead to the greatest improvement in MSE; split the leaf into two child leaves
- Observations
  - In-sample MSE always improves with additional splits
  - What is MSE when each leaf has one observation?

Problem: Tree has been "over-fitted"

- Suppose we fit a tree and pick a particular leaf ℓ.
  - Do we expect that if we drew a new sample, we would get the same answer?

- More formally:
  - Let S^{tr} be training dataset and S^{te} be an independent test set
  - Let $\hat{\mu}(x_i) = \frac{1}{N_{\ell(x_i)}} \sum_{i \in \ell(x_i), S^{tr}} Y_i$
  - Is $E_{i \in S^{te}}[Y_i | X_i \in \ell(x_i)] = \hat{\mu}(x_i)$?

What are tradeoffs in tree depth?

- First: note that in-sample MSE doesn't guide you
  - It always increases with depth

- Tradeoff as you grow tree deeper
  - More personalized predictions
  - More biased estimates

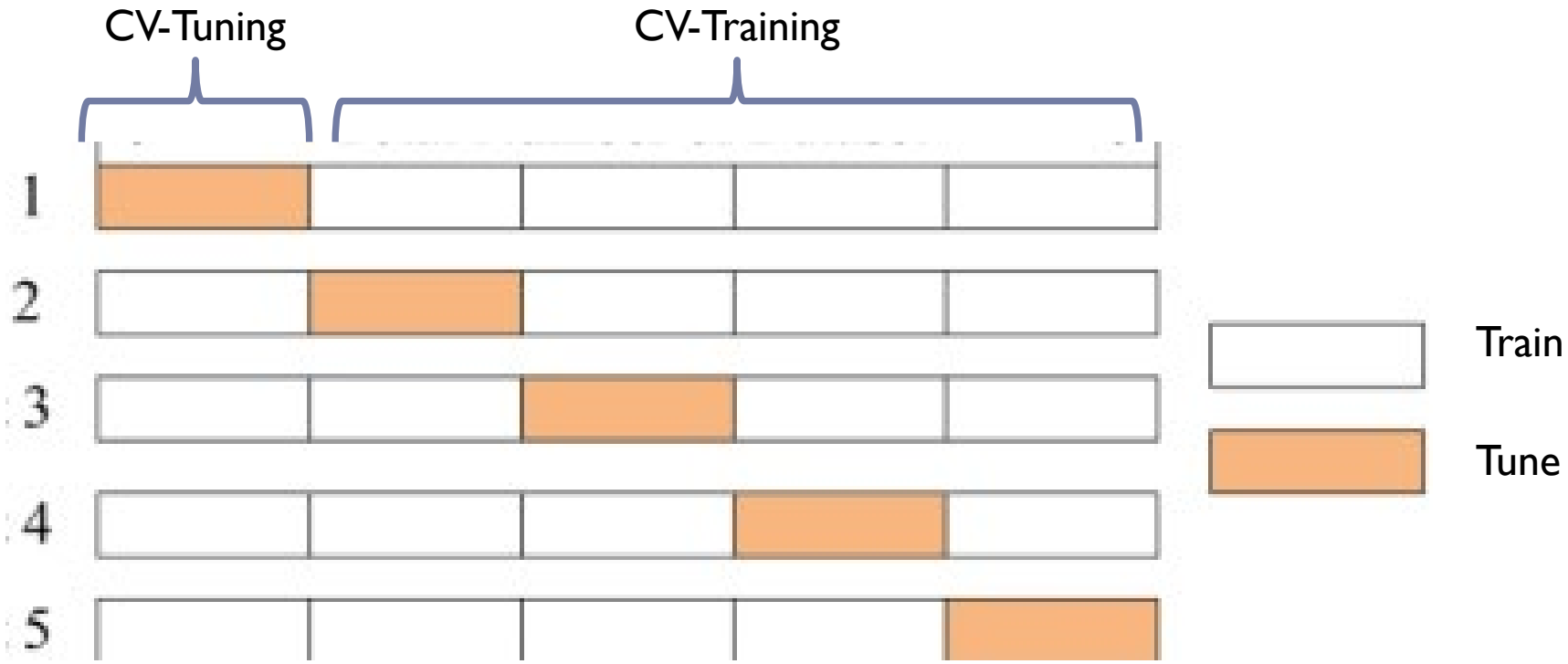
Regression Trees for Prediction: Components

1. Model and Estimation
   A. Model type: Tree structure
   B. Estimator $\hat{\mu}(X_i)$: sample mean of Y_i within leaf $\ell(X_i)$
   C. Set of candidate estimators *C*: correspond to different specifications of how tree is split
2. Criterion function (for fixed tuning parameter λ)
   A. In-sample Goodness-of-fit function:

$$Q^{is} = \text{-MSE (Mean Squared Error)} = -\frac{1}{N}\sum_{i=1}^{N}(\hat{\mu}(X_i) - Y_i)^2$$

   A. Structure and use of criterion
      i. Criterion: $Q^{crit} = Q^{is} - \lambda \times \#$ leaves
      ii. Select member of set of candidate estimators that maximizes Q^{crit}, given λ
3. Cross-validation approach
   A. Approach: Cross-validation on grid of tuning parameters. Select tuning parameter λ with highest Out-of-sample Goodness-of-Fit Q^{os}.
   B. Out-of-sample Goodness-of-fit function: Q^{os} = -MSE

How Does Cross Validation Work?



Tuning Set = 1/5 of Training Set

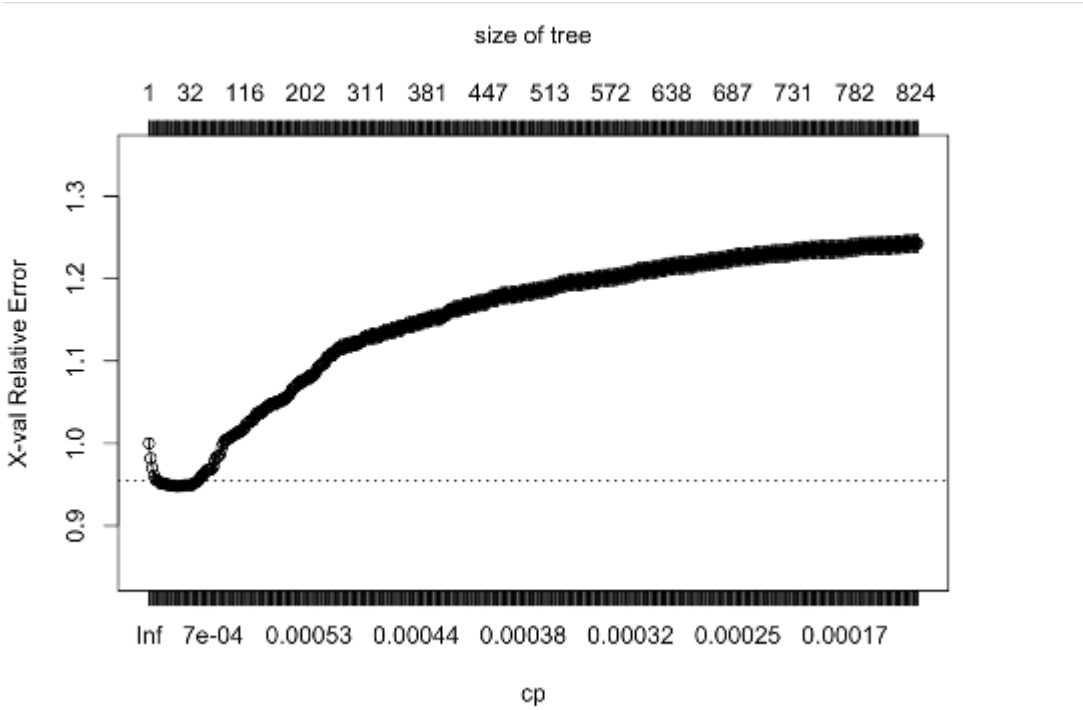
Cross-Validation Mechanics

- Loop over cross-validation samples
  - Train a deep tree on CV-training subset

- Loop over penalty parameters λ
  - Loop over cross-validation samples
    - Prune the tree according to penalty
    - Calculate new MSE of tree
  - Average (over c-v samples) the MSE for this penalty

- Choose the penalty λ^* that gives the best average MSE

Choosing the penalty parameter



Some example code

```
## Regression tree:
## rpart(formula = linear, data = processed.scaled.train, method = "anova",
##     y = TRUE, control = rpart.control(cp = 1e-04, minsplit = 30))
##
## Variables actually used in tree construction:
##  [1] bach_orhigher            city
##  [3] employ_20to64            g2000
##  [5] g2002                    hh_size
##  [7] highschool               median_age
##  [9] median_income            noise1
## [11] noise10                  noise11
## [13] noise12                  noise13
## [15] noise2                   noise3
## [17] noise4                   noise5
## [19] noise6                   noise7
## [21] noise8                   noise9
## [23] p2000                    p2002
## [25] p2004                    percent_62yearsandover
## [27] percent_black            percent_hispanicorlatino
## [29] percent_male             percent_white
## [31] sex                      totalpopulation_estimate
## [33] W                        yob
##
```

```
## Root node error: 3866.8/18000 = 0.21482
##
## n= 18000
##
##             CP nsplit rel error  xerror      xstd
## 1   0.01831622      0   1.00000 1.00020 0.0060337
## 2   0.01200939      1   0.98168 0.98201 0.0061607
## 3   0.00903665      2   0.96967 0.97013 0.0061355
## 4   0.00555973      3   0.96064 0.96125 0.0062722
## 5   0.00296112      4   0.95508 0.95571 0.0061583
## 6   0.00274262      5   0.95212 0.95495 0.0062149
## 7   0.00267924      6   0.94937 0.95394 0.0062370
## 8   0.00190289      7   0.94670 0.95150 0.0062622
## 9   0.00183424      8   0.94479 0.95162 0.0063299
## 10  0.00181651      9   0.94296 0.95154 0.0063322
```

```
## 44  0.00066122     64   0.89338 0.98640 0.0074692
## 45  0.00064984     67   0.89135 0.99433 0.0076063
## 46  0.00064533     68   0.89070 0.99997 0.0077120
## 47  0.00063905     71   0.88876 1.00373 0.0077753
## 48  0.00063765     72   0.88813 1.00493 0.0078130
## 49  0.00063654     78   0.88429 1.00529 0.0078222
## 50  0.00063212     85   0.87957 1.00727 0.0078509
## 51  0.00063205     86   0.87893 1.00815 0.0078690
## 52  0.00062566     94   0.87385 1.00952 0.0078949
## 53  0.00062404     96   0.87260 1.01128 0.0079362
## 54  0.00062352     99   0.87073 1.01200 0.0079494
## 55  0.00061992    102   0.86886 1.01396 0.0079794
## 56  0.00061970    103   0.86824 1.01481 0.0079986
## 57  0.00061887    105   0.86700 1.01494 0.0080002
## 58  0.00061518    112   0.86228 1.01661 0.0080294
```

Pruning Code

```
op.index <- which.min(linear.singletree$cptable[, "xerror"])
cp.vals <- linear.singletree$cptable[, "CP"]
treepruned.linearsingle <- prune(linear.singletree, cp = cp.vals[op.index])
```

A Basic Policy Problem

- Every transfer program in the world must determine…
  - Who is eligible for the transfer
- Typical goal of redistributive programs
  - Transfer to neediest
- But identifying the neediest is easier said than done

Thanks to Sendhil Mullainathan for providing this worked out example….

Typical Poverty Scorecard

Indicator	Value	Points	Sco
1. How many members does the household have?	A. Five or more	0	
	B. Four	6	
	C. Three	11	
	D. Two	17	
	E. One	20	
2. Do any household members ages 5 to 18 go to private school or private pre-school?	A. No	0	
	B. Yes	5	
	C. No members ages 5 to 18	7	
3. How many years of schooling has the female head/spouse completed?	A. Three or less	0	
	B. Four to eleven	2	
	C. Twelve or more	8	
	D. No female head/spouse	8	
4. How many household members work as employees with a written contract, as civil servants for the government, or in the military?	A. None	0	
	B. One	4	
	C. Two or more	13	
5. In their main occupation, how many household members are managers, administrators, professionals in the arts and sciences, mid-level technicians, or clerks?	A. None	0	
	B. One or more	8	
6. How many rooms does the residence have?	A. One to four	0	
	B. Five	2	
	C. Six	5	
	D. Seven	7	
	E. Eight or more	11	
7. How does the household dispose of sewage?	A. Ditch, other, or no bathroom	0	
	B. Simple hole, or directly into river, lake, or ocean	2	
	C. Septic tank not connected to public sewage/rainwater system	3	
	D. Septic tank connected to public sewage/rainwater system	4	
	E. Direct connection to public sewage/rainwater system	5	
8. Does the household have a refrigerator?	A. No	0	
	B. Yes, with one door	5	
	C. Yes, with two doors	10	
9. Does the household have a washing machine?	A. No	0	
	B. Yes	7	
10. Does the household have a cellular or land-line telephone?	A. None	0	
	B. Cellular but not land-line	5	
	C. Land-line but not cellular	6	
	D. Both	11	

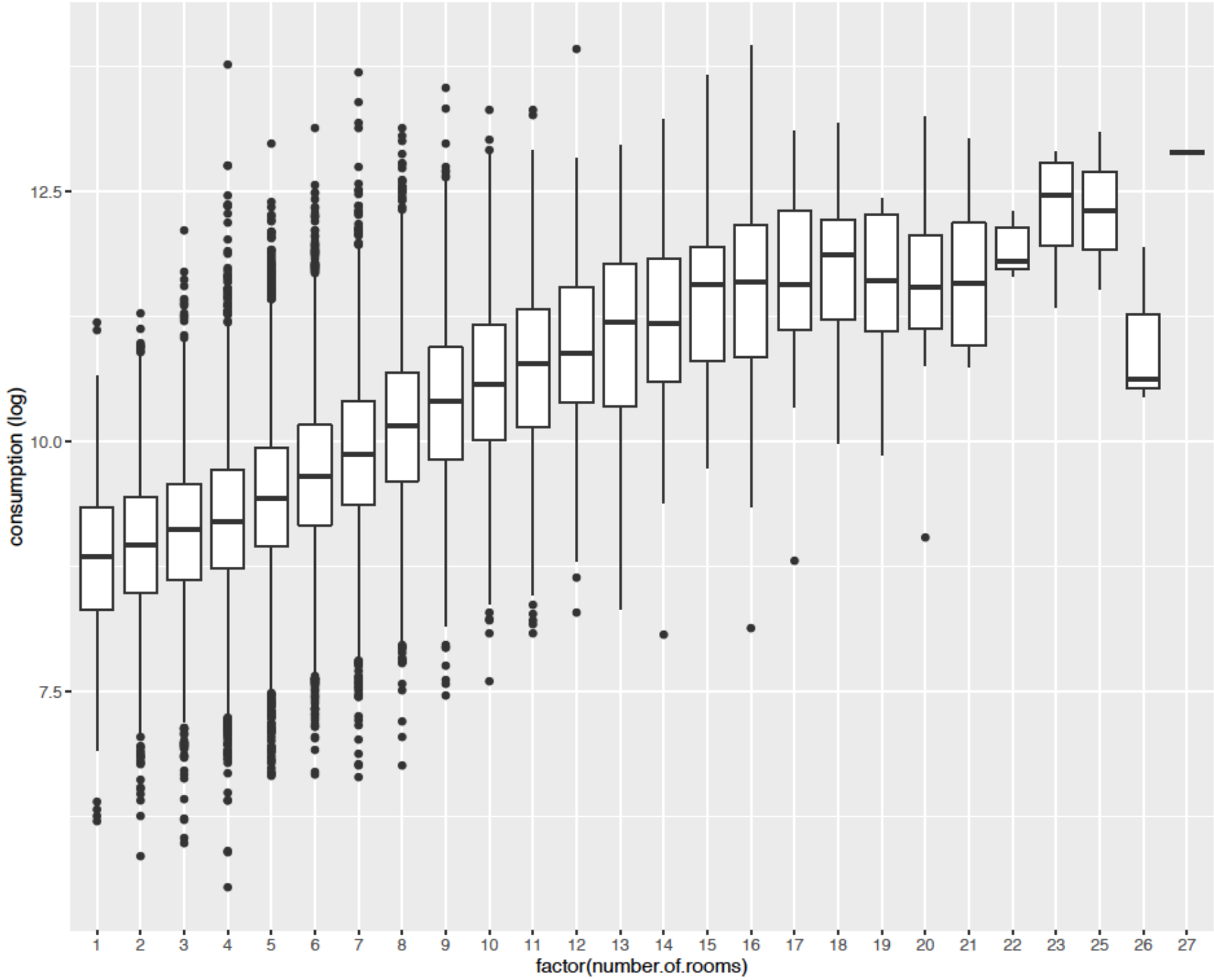
PPI Score	$2.50/Day/2005 PPP Poverty Line	
	Total Below the $2.50/Day/2005 PPP Line	Total Above the $2.50/Day/2005 PPP Line
0-4	81.8%	18.2%
5-9	77.8%	22.2%
10-14	66.1%	33.9%
15-19	49.0%	51.0%
20-24	37.2%	62.8%
25-29	23.9%	76.1%
30-34	15.4%	84.6%
35-39	8.6%	91.4%
40-44	5.2%	94.8%
45-49	3.2%	96.8%
50-54	2.1%	97.9%
55-59	1.2%	98.8%
60-64	1.2%	98.8%
65-69	0.4%	99.6%
70-74	0.6%	99.4%
75-79	0.0%	100.0%
80-84	0.0%	100.0%
85-89	0.0%	100.0%
90-94	0.0%	100.0%
95-100	0.0%	100.0%

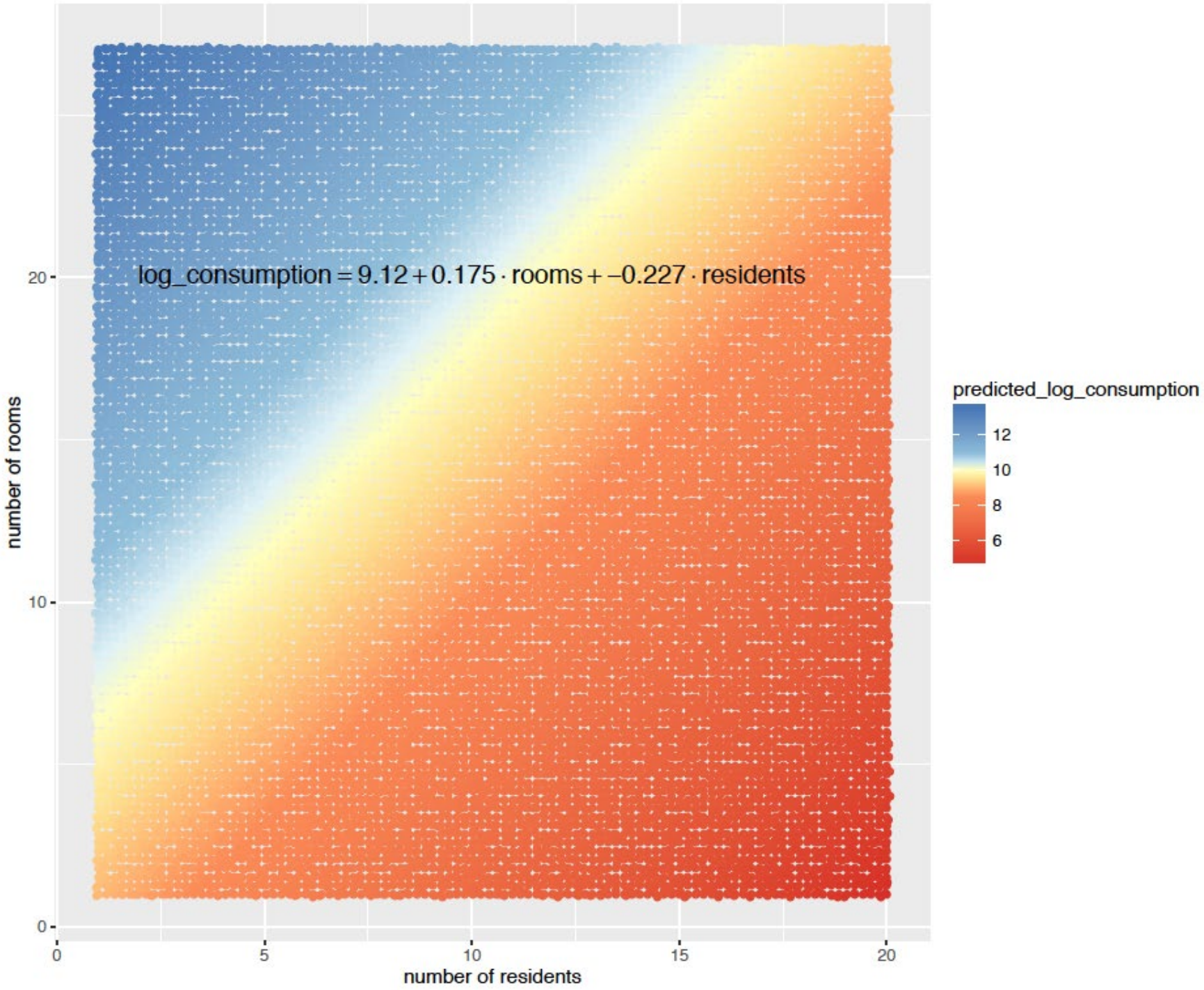
Can we do better?

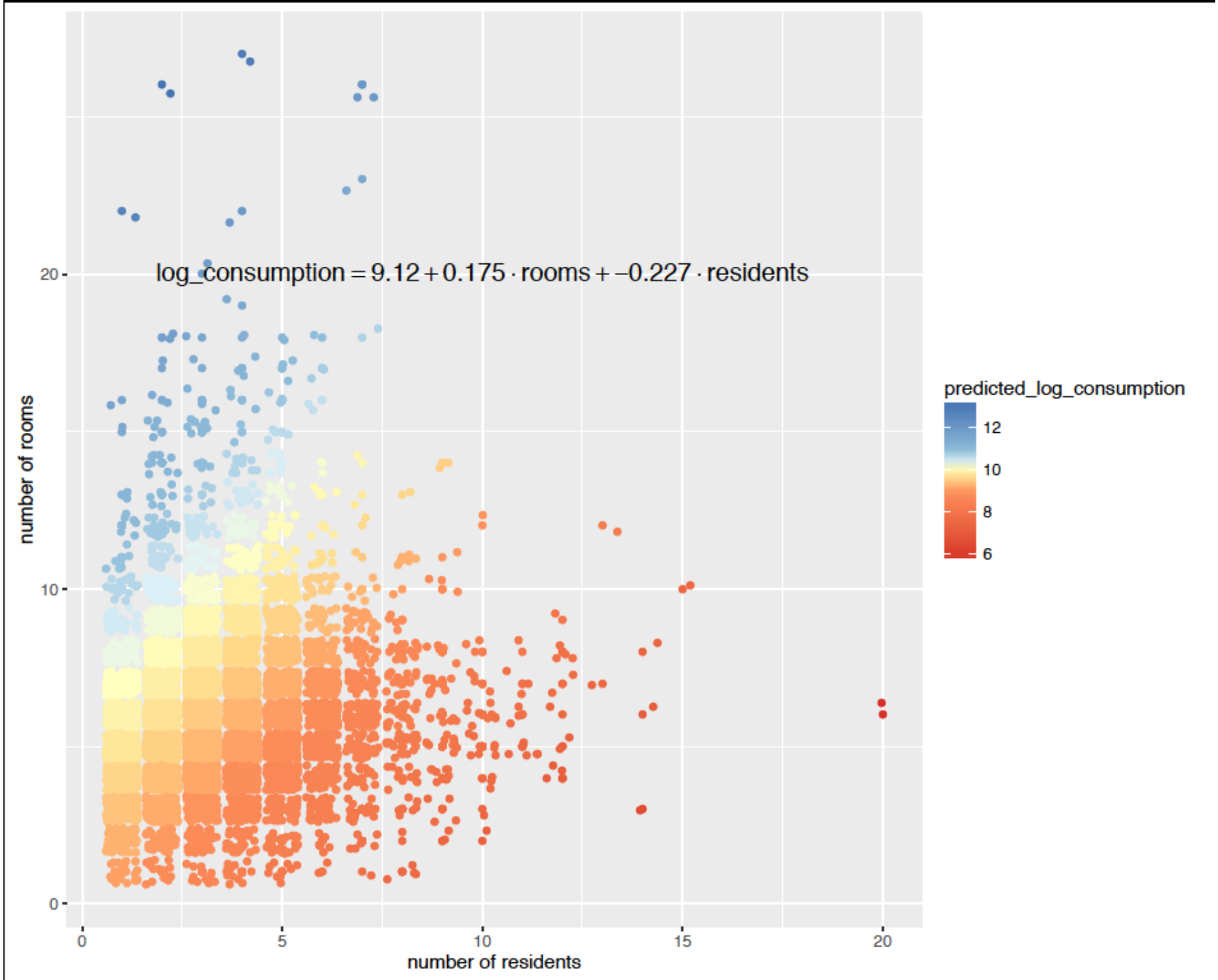
- This component of targeting is a pure prediction problem
- We fundamentally care about getting best predictive accuracy
- Let's use this example to illustrate the mechanics of prediction

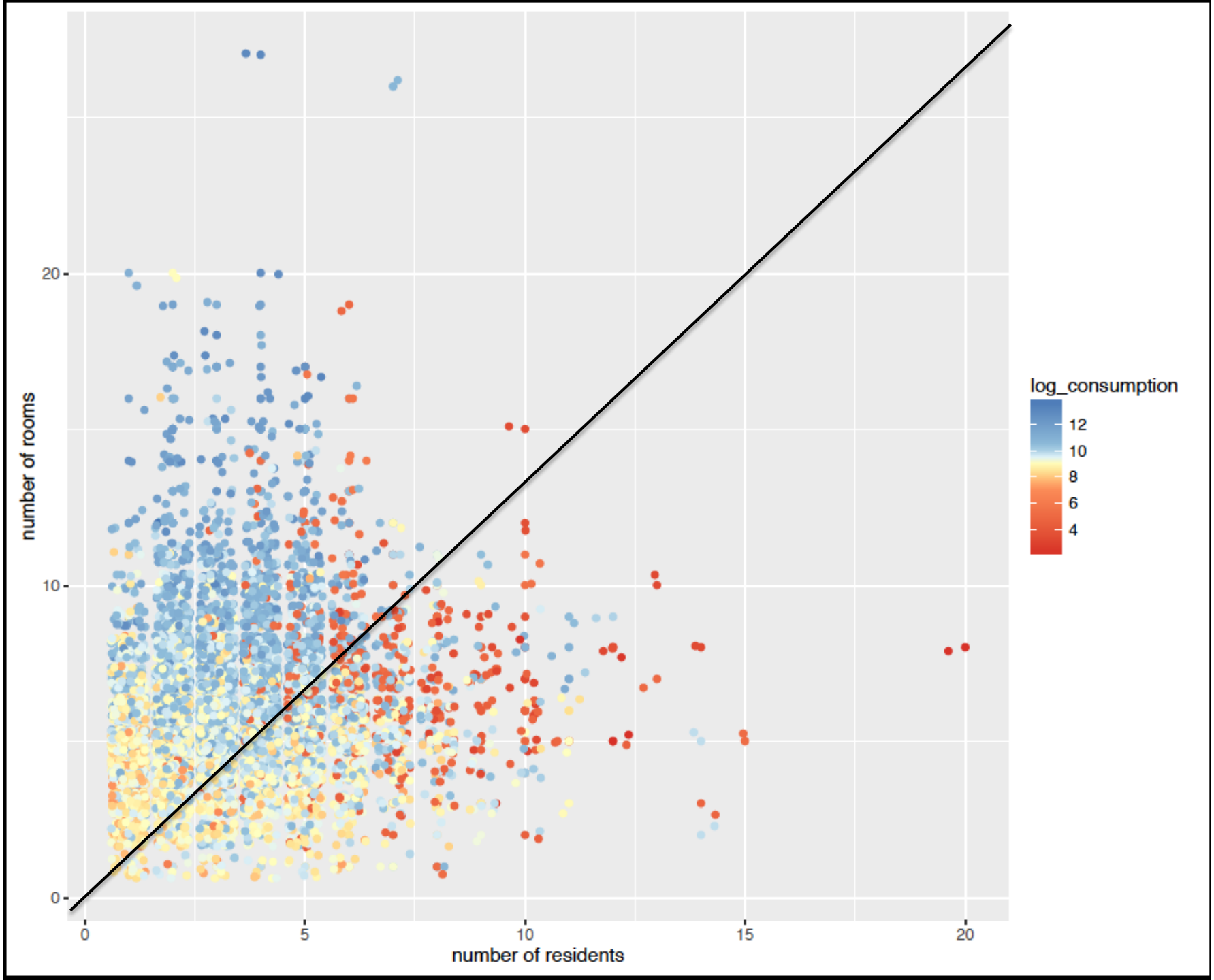
Brazilian Data

- The data:
  - 44,787 data points
  - 53 variables
  - Not very wide?
- Median
  - Annual consumption (in dollars): 3918
  - 348.85 monthly income
- 6 percent below 1.90 poverty line
- 14 percent below the 3.10 poverty line

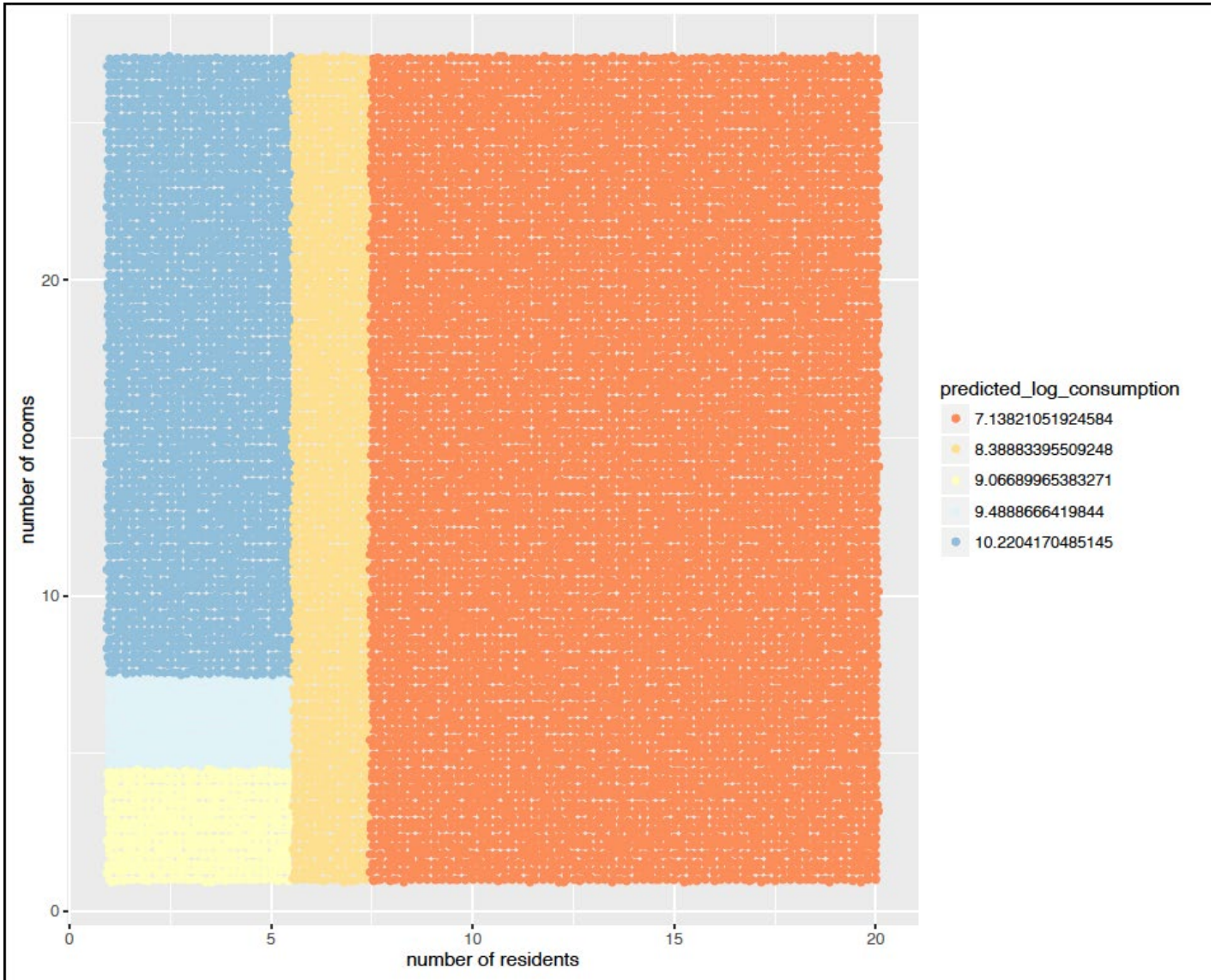

consumption (log)
12.5
10.0
7.5
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
25
26
27
factor(number.of.rooms)

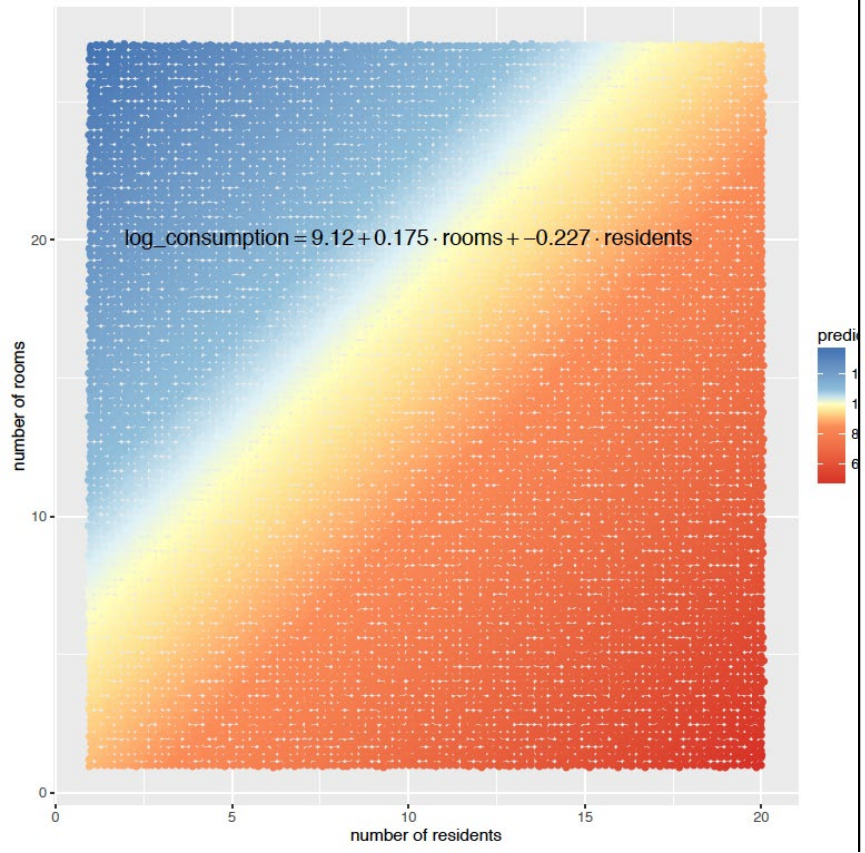

log_consumption = 9.12 + 0.175 · rooms + −0.227 · residents
number of rooms
number of residents
0
10
20
0
5
10
15
20
predicted_log_consumption
12
10
8
6

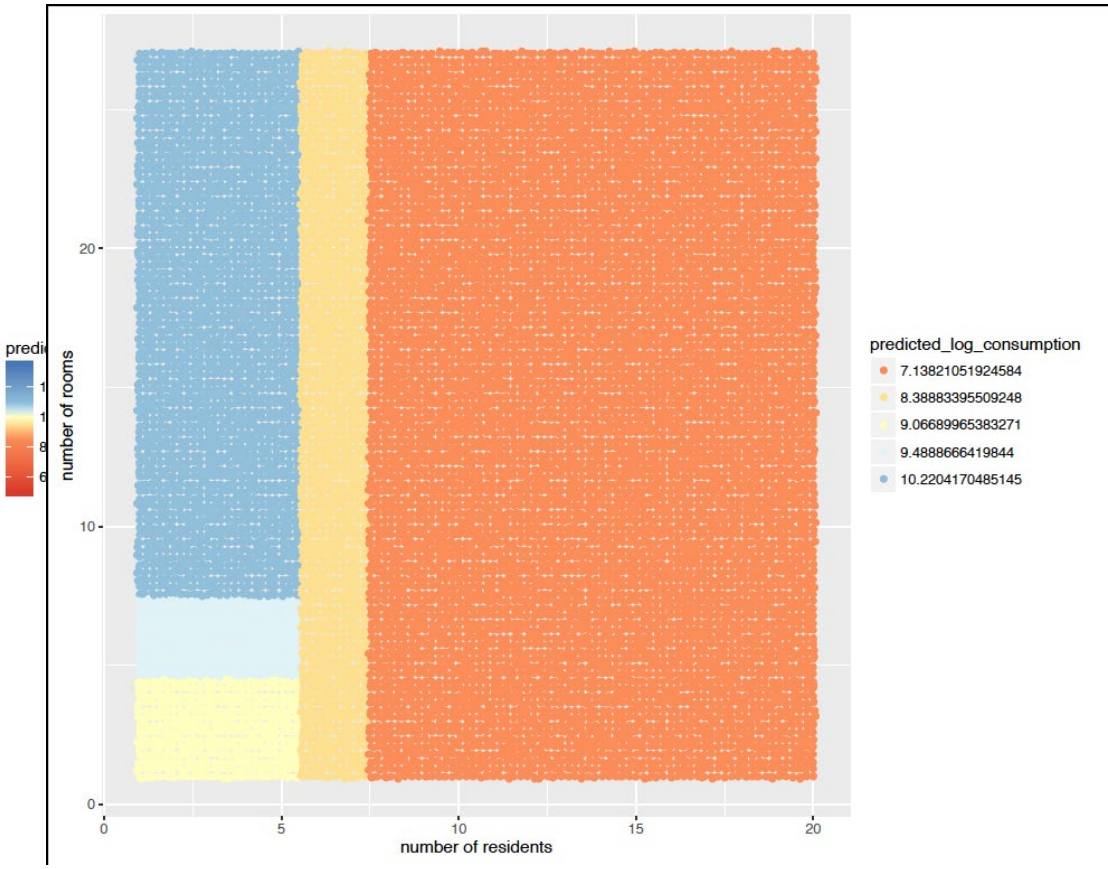

log_consumption = 9.12 + 0.175 · rooms + −0.227 · residents
number of rooms
number of residents
0
10
20
0
5
10
15
20
predicted_log_consumption
12
10
8
6

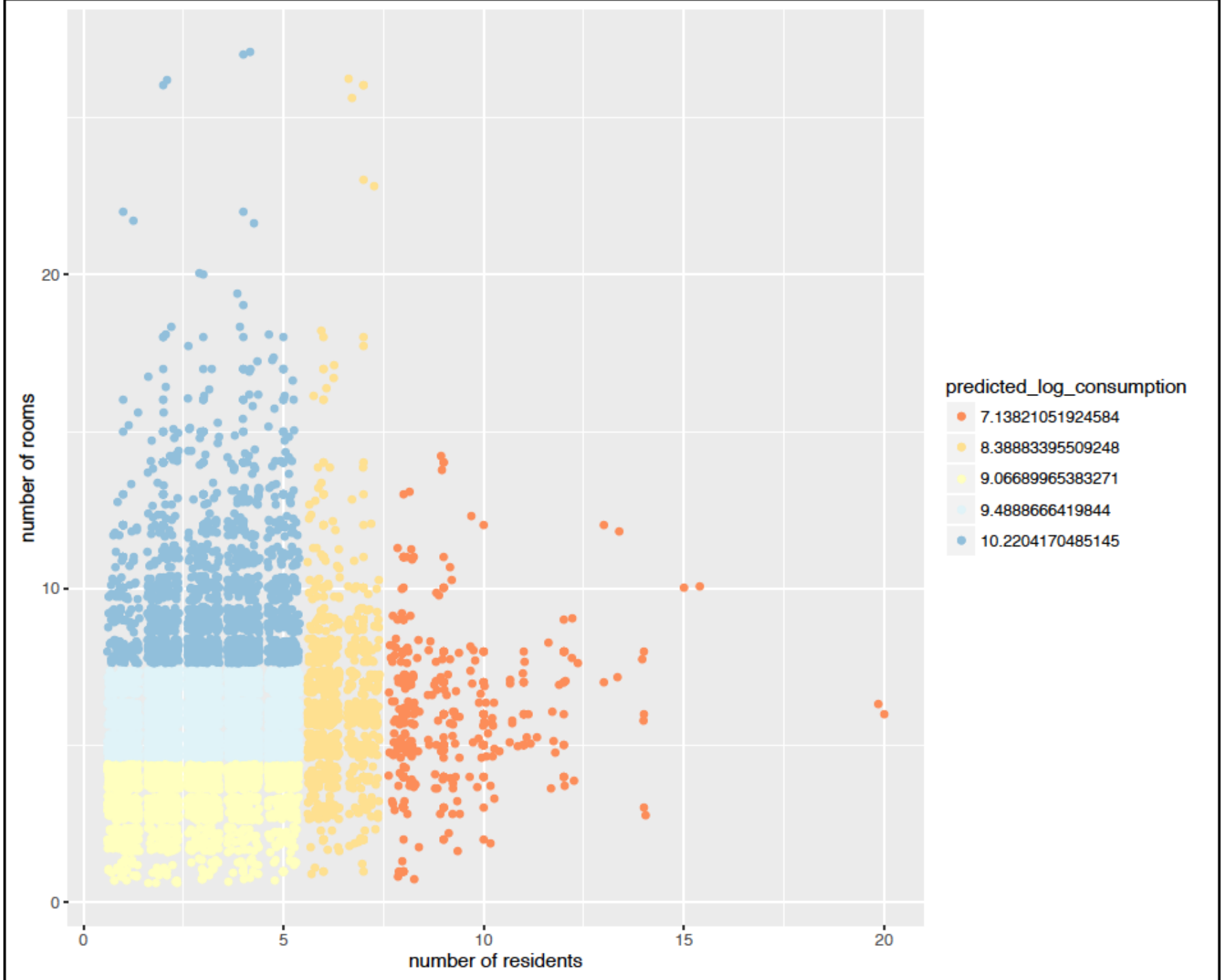

number of rooms
20
10
0
0
5
10
15
20
number of residents
log_consumption
12
10
8
6
4

Two Variable Tree

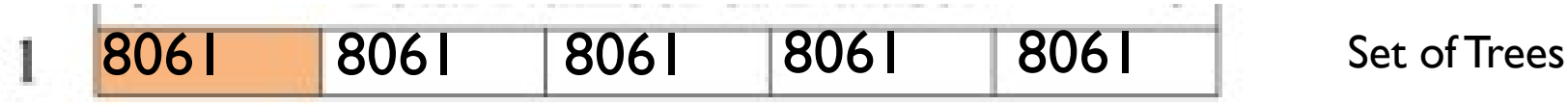



log_consumption = 9.12 + 0.175 · rooms + −0.227 · residents
number of rooms
number of residents
0
5
10
15
20


number of rooms
number of residents
predicted_log_consumption
7.13821051924584
8.38883395509248
9.06689965383271
9.4888666419844
10.2204170485145


number of rooms
number of residents
0
10
20
0
5
10
15
20
predicted_log_consumption
7.13821051924584
8.38883395509248
9.06689965383271
9.4888666419844
10.2204170485145

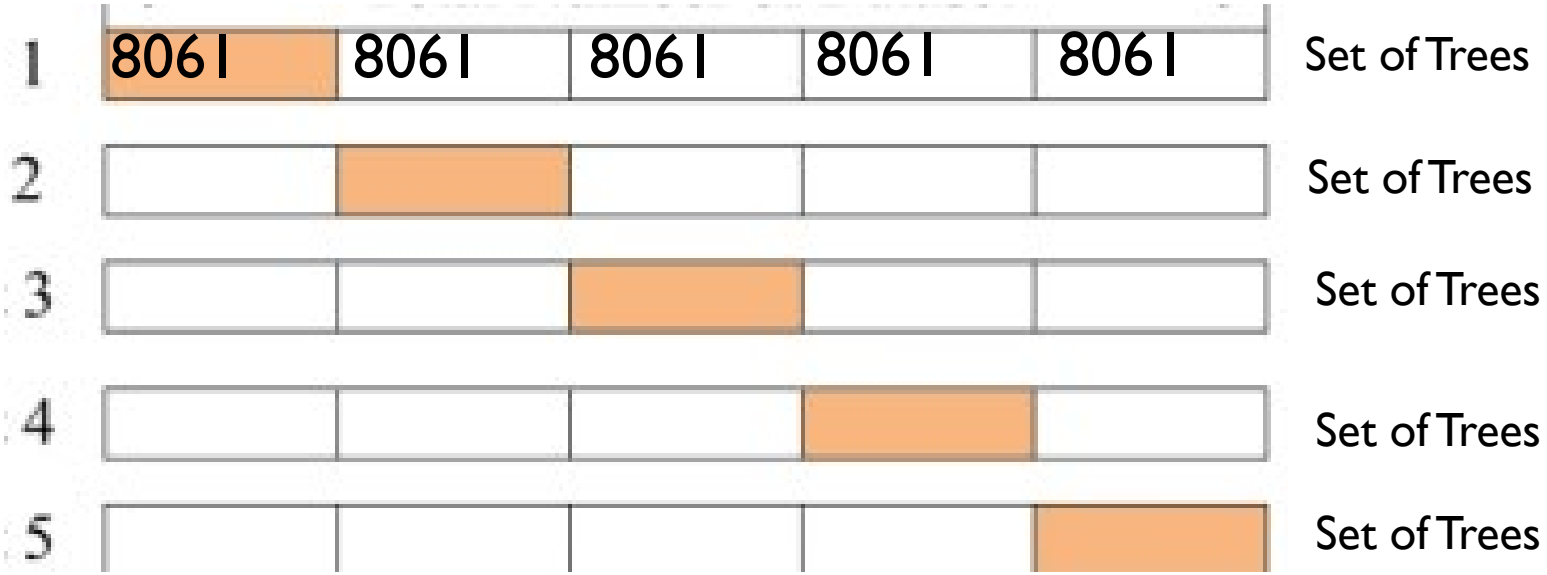
28,573 data points to Fit with



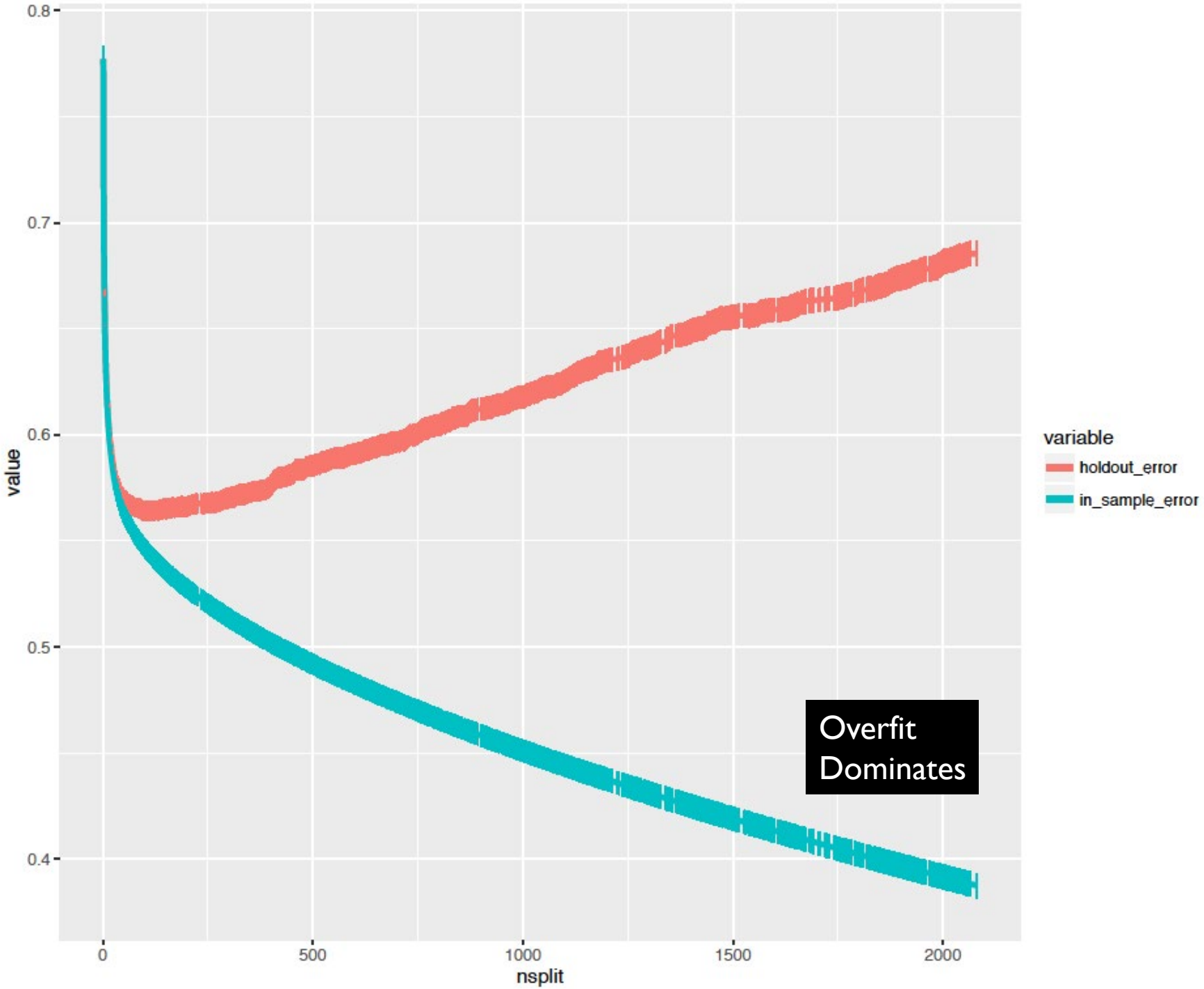
Fit trees on 4/5 of the data

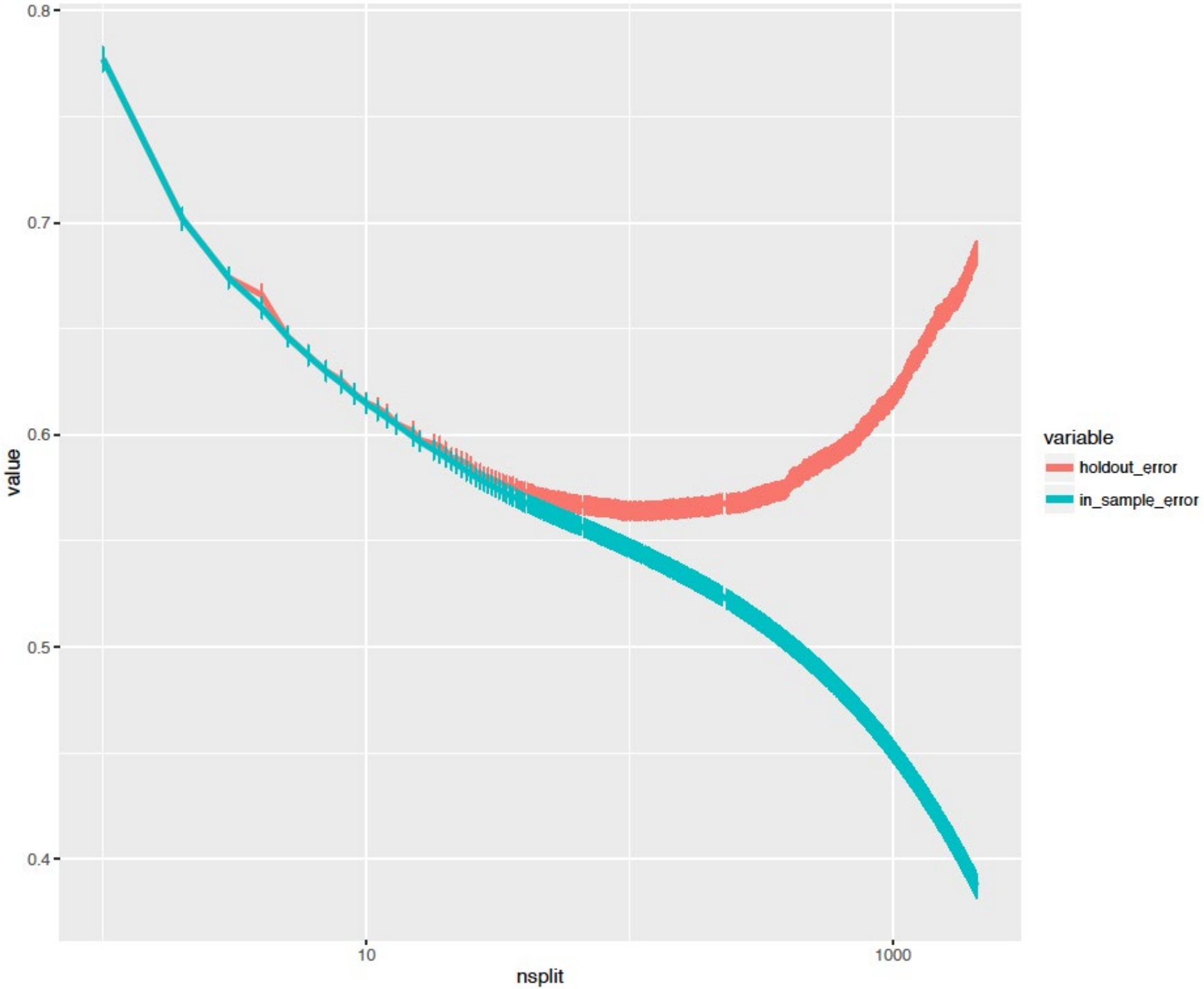
Fit a tree for every level of split size

28,573 data points to Fit with



REPEAT leaving each fold out


0.8
0.7
0.6
0.5
0.4
value
0
500
1000
1500
2000
nsplit
variable
holdout_error
in_sample_error
Overfit
Dominates


0.8
0.7
0.6
0.5
0.4
value
10
1000
nsplit
variable
holdout_error
in_sample_error

Tuning Parameter Choice

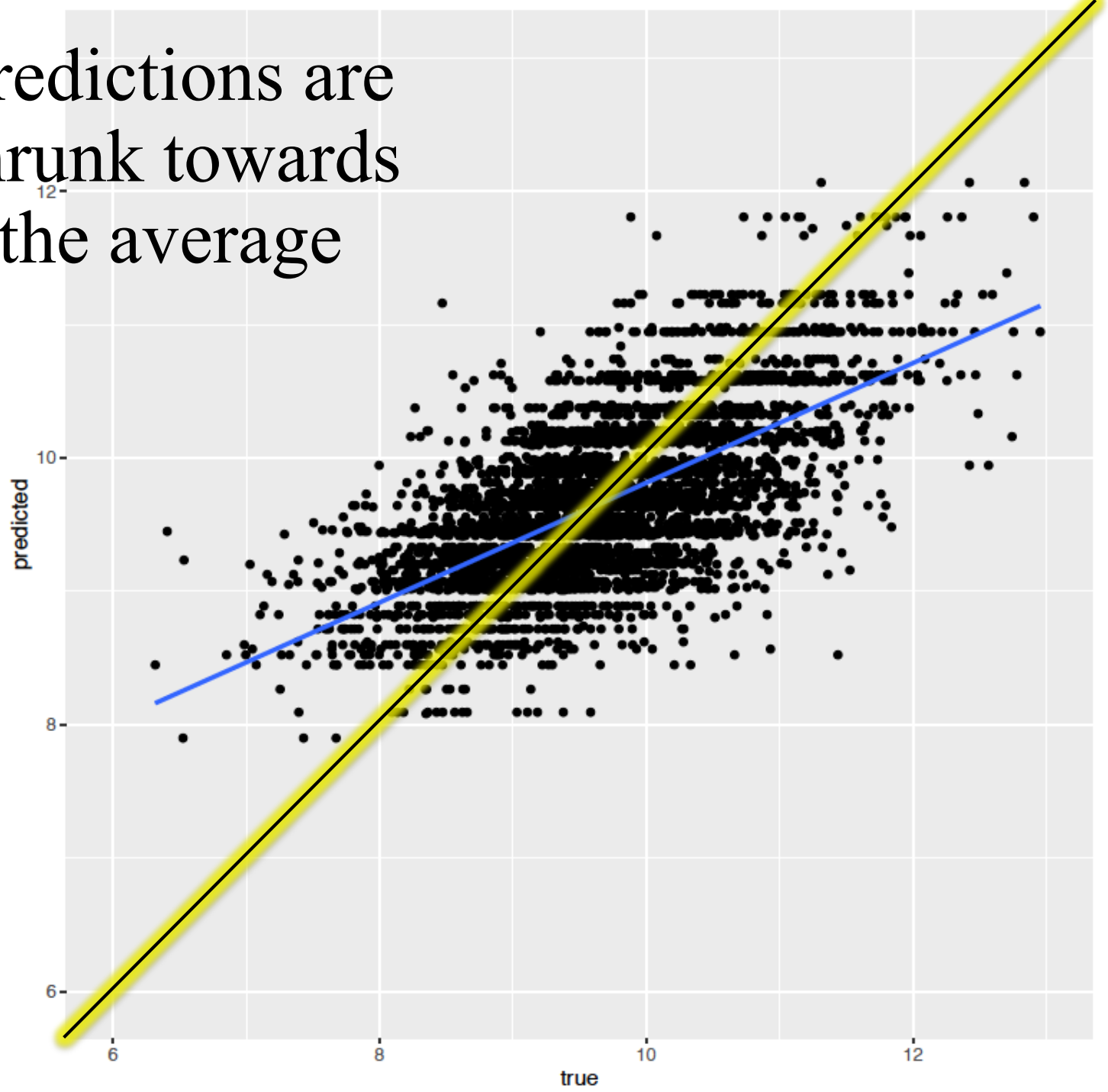
- Minimum?

- One standard error "rule" (rule of thumb)
  - Which direction?

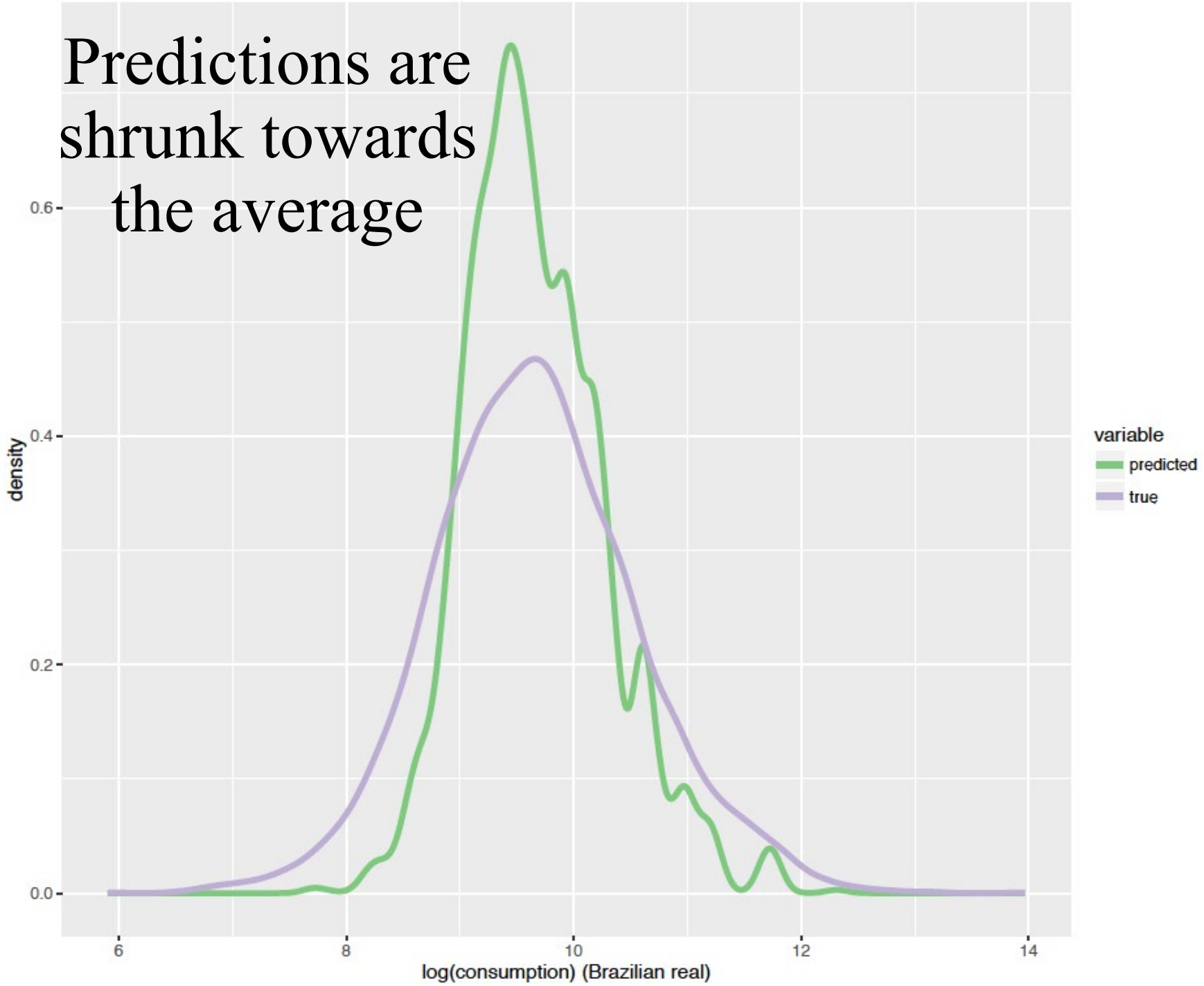
Output

- Which of these many trees do we output?
- Even after choosing lambda we have as many trees as folds…
- Estimate one tree on full data using chosen cut size
- Key point: Cross validation is just for choosing tuning parameter
  - Just for deciding how complex a model to choose

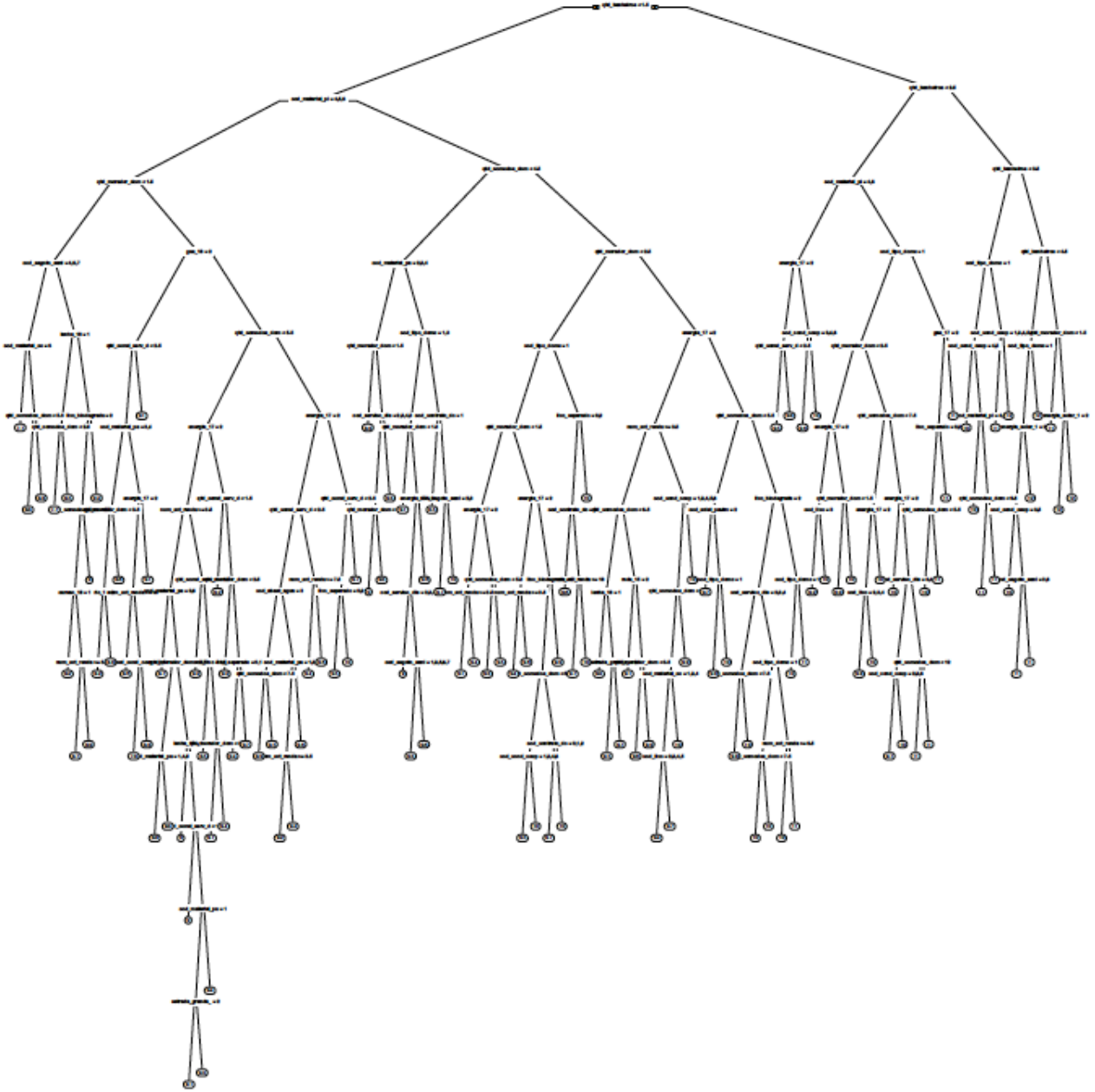
Predictions are shrunk towards the average



Predictions are shrunk towards the average



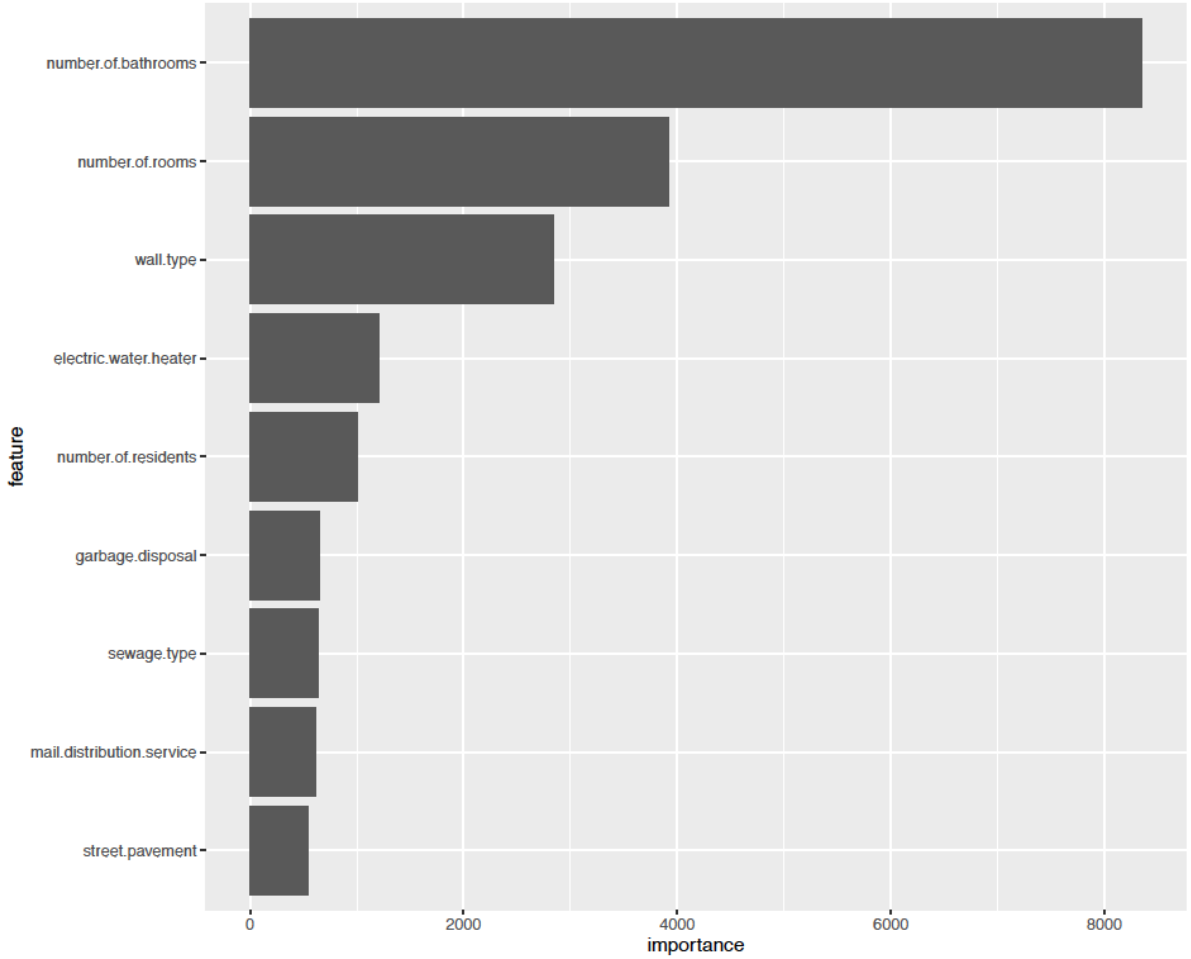
What does the tree look like?



- What else can we look at to get a sense of what the predictions are?

Variable Importance

Empirical loss by noising up *x* minus Empirical loss



How to describe model

- Large discussion of "interpretability"
  - Will return to this
  - Should produce the mean attributes of observations in each leaf, rather than interpret tree structure
- But one implication is that the prediction function itself becomes a new outcome variable to analyze.
- Is any of this stable? What would a confidence interval look like?

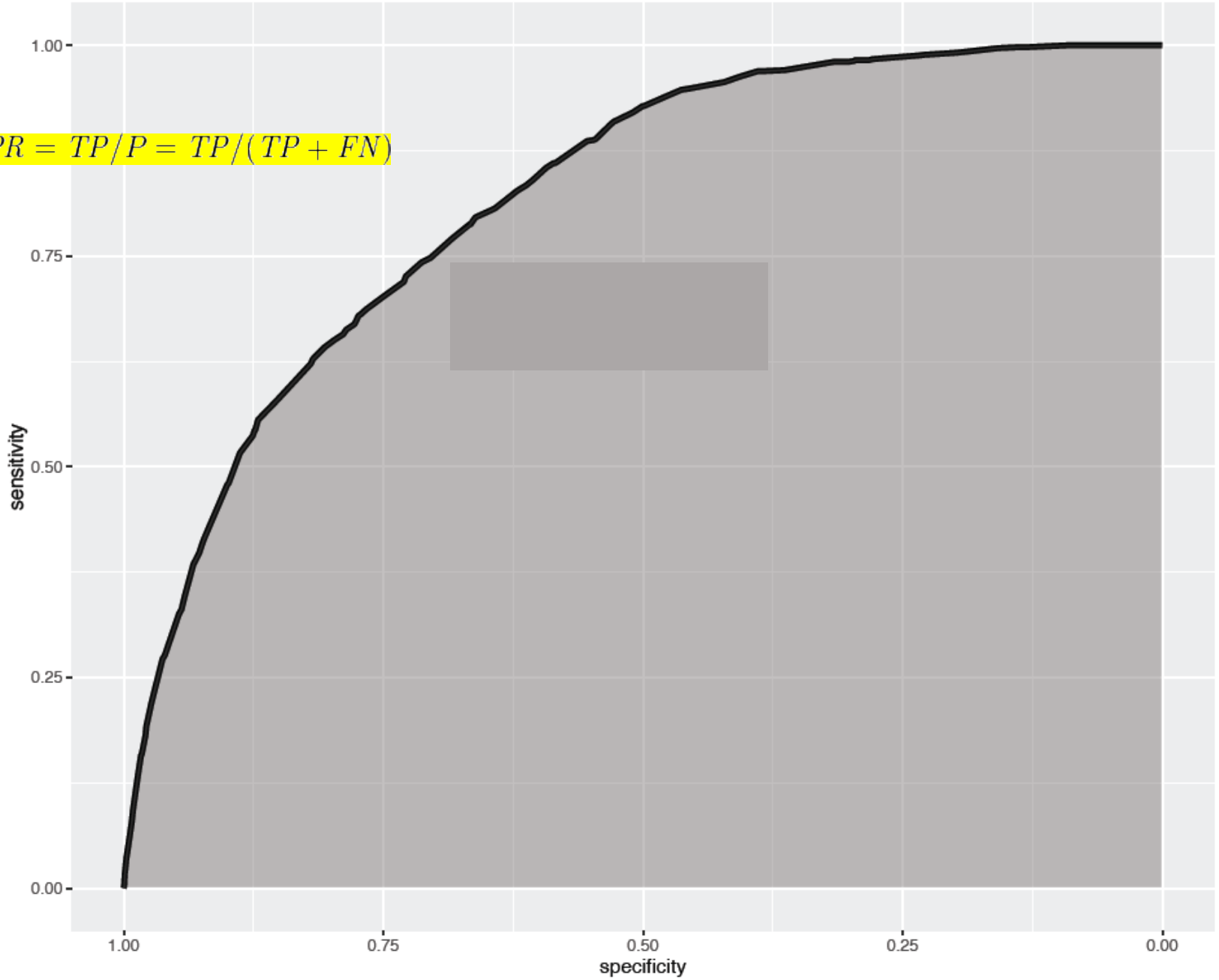
Questions and Observations

- How do we choose hold-out set size?
- How to choose the # of folds?
- What to tune on? (regularizer)
- Which tuning parameter to choose from cross-validation?

- What is stable/robust about the estimated function?

Measuring Performance

		Predicted condition			
	Total population	Predicted Condition positive	Predicted Condition negative	Prevalence $= \frac{\Sigma \text{ Condition positive}}{\Sigma \text{ Total population}}$	
True condition	condition positive	**True positive**	**False Negative** (Type II error)	True positive rate (TPR), Sensitivity, Recall $= \frac{\Sigma \text{ True positive}}{\Sigma \text{ Condition positive}}$	False negative rate (FNR), Miss rate $= \frac{\Sigma \text{ False negative}}{\Sigma \text{ Condition positive}}$
	condition negative	**False Positive** (Type I error)	**True negative**	False positive rate (FPR), Fall-out $= \frac{\Sigma \text{ False positive}}{\Sigma \text{ Condition negative}}$	True negative rate (TNR), Specificity (SPC) $= \frac{\Sigma \text{ True negative}}{\Sigma \text{ Condition negative}}$
	Accuracy (ACC) $= \frac{\Sigma \text{ True positive} + \Sigma \text{ True negative}}{\Sigma \text{ Total population}}$	Positive predictive value (PPV), Precision $= \frac{\Sigma \text{ True positive}}{\Sigma \text{ Test outcome positive}}$	False omission rate (FOR) $= \frac{\Sigma \text{ False negative}}{\Sigma \text{ Test outcome negative}}$	Positive likelihood ratio (LR+) $= \frac{\text{TPR}}{\text{FPR}}$	Diagnostic odds ratio (DOR) $= \frac{\text{LR+}}{\text{LR−}}$
		False discovery rate (FDR) $= \frac{\Sigma \text{ False positive}}{\Sigma \text{ Test outcome positive}}$	Negative predictive value (NPV) $= \frac{\Sigma \text{ True negative}}{\Sigma \text{ Test outcome negative}}$	Negative likelihood ratio (LR−) $= \frac{\text{FNR}}{\text{TNR}}$	



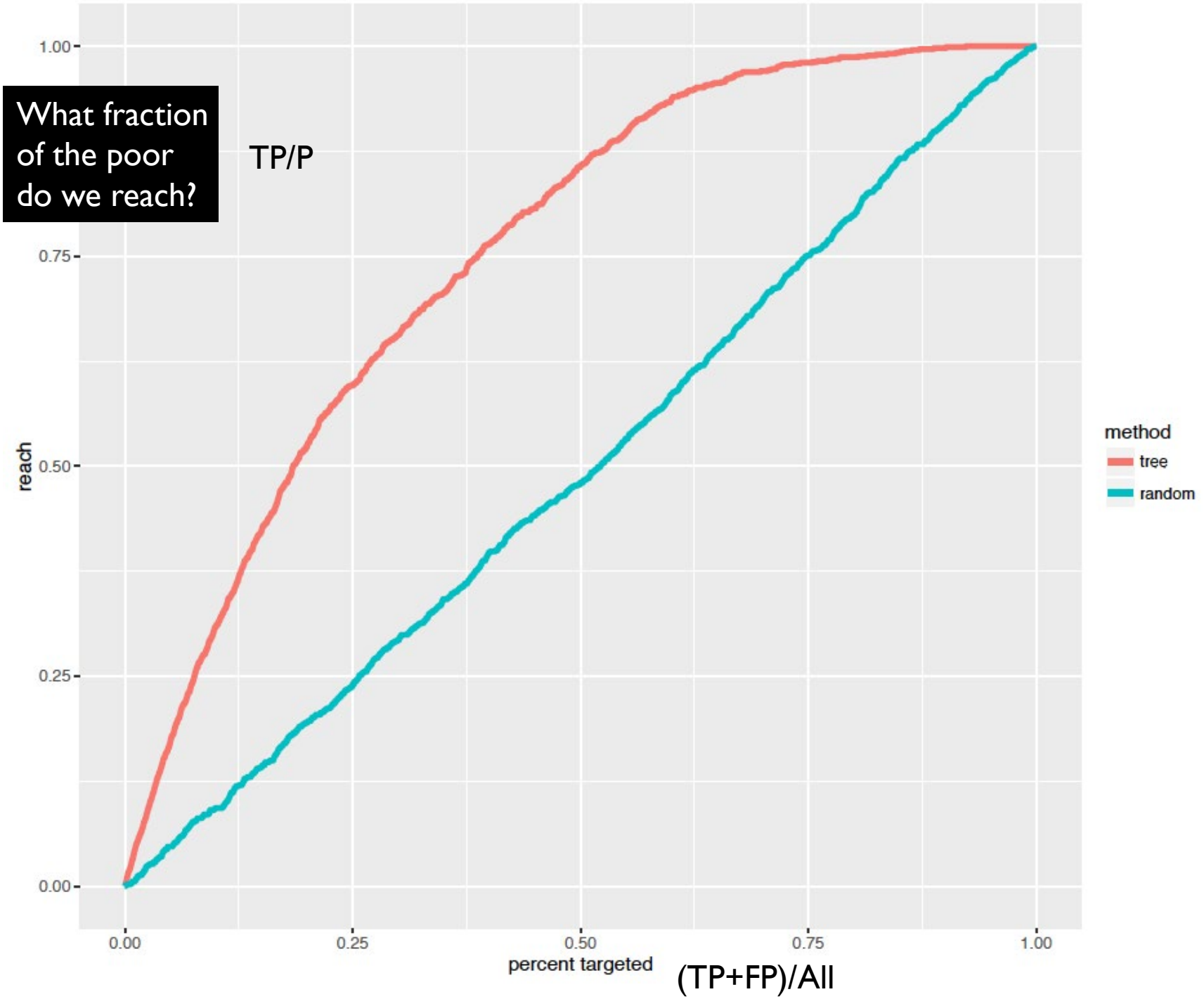
$$TPR = TP/P = TP/(TP + FN)$$

$$SPC = TN/N = TN/(TN + FP)$$

Measuring Performance

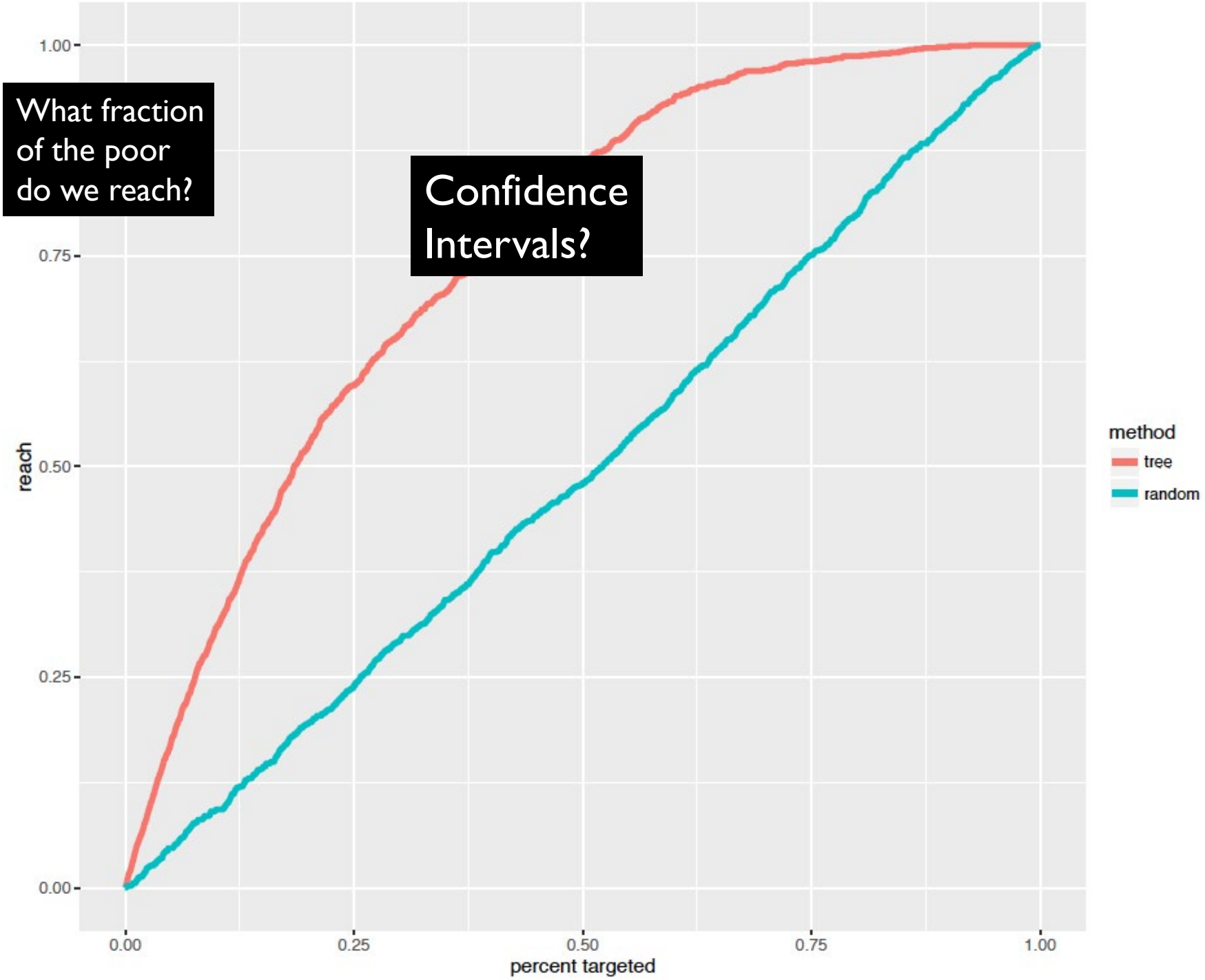
- Area Under Curve: Typical measure of performance

- Does this measure capture the right economic question?


What fraction of the poor do we reach?
TP/P
1.00
0.75
0.50
0.25
0.00
reach
0.00
0.25
0.50
0.75
1.00
percent targeted
(TP+FP)/All
method
tree
random

Measuring Performance

- AUC: Typical measure of performance
- What do you think of this measure?
- Getting the domain specific meaningful performance measure
  - **Magnitudes**
  - **Need point of comparison**


What fraction of the poor do we reach?
Confidence Intervals?
1.00
0.75
0.50
0.25
0.00
reach
0.00
0.25
0.50
0.75
1.00
percent targeted
method
tree
random

Theoretical Guidance?

- How do we choose hold-out set size?
- How to choose the # of folds?
- What to tune on? (regularizer)
- Which tuning parameter to choose from cross-validation?
- What is stable/robust about the estimated function?
- How do we form standard errors on performance?

Summary

- Regression trees easy to understand and interpret
  - DON'T MISINTERPRET
  - Just because tree doesn't split (LASSO doesn't select), does not mean that a variable is not important
  - Consider the characteristics of the subgroup
- Tradeoff between personalized versus inaccurate predictions
- Cross-validation is a tool to figure out the best balance in a particular dataset
  - E.g if truth is complex, may want to go deeper
- CART is ad hoc, but works well in practice
  - Loses to OLS/logit if true model is linear
  - Good at finding lots of complex interactions

Causal Trees: Recursive Partitioning for Heterogeneous Causal Effects

Motivation I: Experiments and Data-Mining

- Concerns about ex-post "data-mining"
  - In medicine, scholars required to pre-specify analysis plan
  - In economic field experiments, calls for similar protocols
- But how is researcher to predict all forms of heterogeneity in an environment with many covariates?
- Goal:
  - Allow researcher to specify set of potential covariates
  - Data-driven search for heterogeneity in causal effects with valid standard errors

Motivation II: Treatment Effect Heterogeneity for Policy

- Estimate of treatment effect heterogeneity needed for optimal decision-making
- This paper focuses on estimating treatment effect as function of attributes directly, not optimized for choosing optimal policy in a given setting
- This "structural" function can be used in future decision-making by policy-makers without the need for customized analysis

Preview

- Distinguish between causal effects and attributes
- Estimate treatment effect heterogeneity:
  - Introduce estimation approaches that combine ML prediction & causal inference tools
- Introduce and analyze new cross-validation approaches for causal inference
- Inference on estimated treatment effects in subpopulations
  - Enabling post-experiment data-mining

- NOTE: estimation versus prediction objective

"Moving the Goalpost": What is Question?

- Estimate $\tau(x) = E[\tau_i | X_i = x]$ as well as possible
  - Why? Want to hold some covariates fixed and look at the effect of others.
- Estimate $\text{BLP}[\tau_i | X_i = x]$
  - Why? "Interpretable"? The best linear predictor is a bit hard to interpret without the whole variance-covariance matrix of nonlinear functions and interactions; you have omitted variable bias on the coefficients you are explaining, relative to $\tau(x)$. My view is that simple models can be more "mis-interpretable" than interpretable.
- Causal Tree: Find partition of covariate space and estimate $E[\tau_i | X_i \in S]$ for each element of partition
  - Why? Easier to interpret than BLP, but still important to report mean, median, percentiles of all covariates for each leaf to understand how leaves are different, when covariates are correlated.
- Which units have highest or lowest treatment effects?
  - Why? Helps understand who could be treated. Can be estimated directly or can draw inferences based on output of causal tree or non-parametric estimates of $\tau(x)$
  - Common practice to display differences between covariates; see Chernozhukov and Duflo (2018)
- What is the best policy mapping from X to treatments W?
  - Why? Sometimes this is the direct object of interest.
  - Fully nonparametric? See e.g. Hirano and Porter (2009)
  - With limited complexity or other constraints? See e.g. Kitagawa and Tetenov (2015), Athey and Wager (2017).
- What is the full set of covariates for which there is statistically significant heterogeneity?
  - List, Shaikh, and Xu (2016) (multiple testing)
- Tradeoffs: More personalization, reliable confidence intervals, role of assumptions, interpretability

Using Trees to Estimate Causal Effects

Model:

$$Y_i \quad = Y_i(W_i\,) \quad = \begin{cases} Y_i(1) & if \;\; W_i = 1, \\ Y_i(0) & otherwise. \end{cases}$$

- Suppose random assignment of W_i

- Want to predict individual *i*'s treatment effect
  - $\tau_i = Y_i(1) - Y_i(0)$
  - This is not observed for any individual
  - Not clear how to apply standard machine learning tools
- Let

$$\mu(w, x) = \mathbb{E}[Y_i | W_i = w, X_i = x]$$
$$\tau(x) = \mu(1, x) - \mu(0, x)$$

Using Trees to Estimate Causal Effects

$$\mu(w, x) = \mathbb{E}[Y_i | W_i = w, X_i = x]$$
$$\tau(x) = \mu(1, x) - \mu(0, x)$$

- Approach 1: Analyze two groups separately
  - Estimate $\hat{\mu}(1, x)$ using dataset where $W_i = 1$
  - Estimate $\hat{\mu}(0, x)$ using dataset where $W_i = 0$
  - Use propensity score weighting (PSW) if needed
  - Do within-group cross-validation to choose tuning parameters
  - Construct prediction using
    $$\hat{\mu}(1, x) - \hat{\mu}(0, x)$$
- Approach 2: Estimate $\mu(w, x)$ using tree including both covariates
  - Include PS as attribute if needed
  - Choose tuning parameters as usual
  - Construct prediction using
    $$\hat{\mu}(1, x) - \hat{\mu}(0, x)$$
  - Estimate is zero for *x* where tree does not split on *w*

- Observations
  - Estimation and cross-validation not optimized for goal
  - Lots of segments in Approach 1: combining two distinct ways to partition the data

- Problems with these approaches
  1. Approaches not tailored to the goal of estimating treatment effects
  2. How do you evaluate goodness of fit for tree splitting and cross-validation?
     - $\tau_i = Y_i(1) - Y_i(0)$ is not observed and thus you don't have ground truth for any unit

Literature

Approaches in the spirit of single tree and two trees

- Beygelzimer and Langford (2009)
  - Analogous to "two trees" approach with multiple treatments; construct optimal policy
- Dudick, Langford, and Li (2011)
  - Combine inverse propensity score method with "direct methods" (analogous to single tree approach) to estimate optimal policy
- Foster, Tailor, Ruberg, *Statistics and Medicine* (2011)
  - Estimate $\mu(w, x)$ using random forests, define $\hat{\tau}_i = \hat{\mu}(1, X_i) - \hat{\mu}(0, X_i)$, and do trees on $\hat{\tau}_i$.
- Imai and Ratkovic (2013)
  - In context of randomized experiment, estimate $\mu(w, x)$ using lasso type methods, and then $\hat{\tau}(x) = \hat{\mu}(1, x) - \hat{\mu}(0, x)$.

Estimating treatment effects directly at leaves of trees

- Su, Tsai, Wang, Nickerson, Li (2009)
  - Do regular tree, but split if the t-stat for the treatment effect difference is large, rather than when the change in prediction error is large.
- Zeileis, Hothorn, and Hornick (2005)
  - "Model-based recursive partitioning": estimate a model at the leaves of a tree. In-sample splits based on prediction error, do not focus on out of sample cross-validation for tuning.

Transformed outcomes or covariates for regressions

- Tibshirani et al (2014)
- Weisberg and Pontes (2015)

Another Approach: Transform the Outcome

- Suppose we have 50-50 randomization of treatment/control
  - Let $Y_i^* = \begin{cases} 2Y_i & if\ W_i = 1 \\ -2Y_i & if\ W_i = 0 \end{cases}$
  - Then $E[Y_i^*] = 2 \cdot \left(\frac{1}{2}E[Y_i(1)] - \frac{1}{2}E[Y_i(0)]\right) = E[\tau_i]$

- Suppose treatment with probability p_i
  - Let $Y_i^* = \frac{W_i - p}{p(1-p)} Y_i = \begin{cases} \frac{1}{p}Y_i & if\ W_i = 1 \\ -\frac{1}{1-p}Y_i & if\ W_i = 0 \end{cases}$
  - Then $E[Y_i^*] = \left(p\frac{1}{p}E[Y_i(1)] - (1-p)\frac{1}{1-p}E[Y_i(0)]\right) = E[\tau_i]$

- Observational study w/ unconfoundedness or stratified experiment
  - Let $Y_i^* = \frac{W_i - p(X_i)}{p(X_i)(1-p(X_i))} Y_i$
  - Estimate $\hat{p}(x)$ using traditional methods
  - Can also residualize the outcome first, but this introduces add'l dependence on the model for the conditional mean

Causal Trees:
(Conventional Tree, Transformed Outcome)

1. Model and Estimation
   A. Model type: Tree structure
   B. Estimator $\hat{\tau}_i^*$: sample mean of Y_i^* within leaf
   C. Set of candidate estimators *C*: correspond to different specifications of how tree is split
2. Criterion function (for fixed tuning parameter λ)
   A. In-sample Goodness-of-fit function:

$$Q^{is} = \text{-MSE (Mean Squared Error)} = -\frac{1}{N}\sum_{i=1}^{N}(\hat{\tau}_i^* - Y_i^*)^2$$

   A. Structure and use of criterion
      i. Criterion: $Q^{crit} = Q^{is} - \lambda \times \#$ leaves
      ii. Select member of set of candidate estimators that maximizes Q^{crit}, given λ
3. Cross-validation approach
   A. Approach: Cross-validation on grid of tuning parameters. Select tuning parameter λ with highest Out-of-sample Goodness-of-Fit Q^{os}.
   B. Out-of-sample Goodness-of-fit function: Q^{os} = -MSE

Critique of Approach: Transform the Outcome

$$Y_i^* = \frac{W_i - p}{p(1-p)} Y_i = \begin{cases} \frac{1}{p} Y_i & if\ W_i = 1 \\ -\frac{1}{1-p} Y_i & if\ W_i = 0 \end{cases}$$

- Within a leaf, sample average of Y_i^* is not most efficient estimator of treatment effect
  - The proportion of treated units within the leaf is not the same as the overall sample proportion
- This motivates preferred approach: use sample average treatment effect in the leaf

Causal Trees:
(Causal Tree, TOT loss function)

1. Model and Estimation
   A. Model type: Tree structure
   B. Estimator $\hat{\tau}_i^{CT}$: sample average treatment effect within leaf (w/ PSW)
   C. Set of candidate estimators *C*: correspond to different specifications of how tree is split
2. Criterion function (for fixed tuning parameter λ)
   A. In-sample Goodness-of-fit function:

$$Q^{is} = \text{-MSE (Mean Squared Error)} = -\frac{1}{N}\sum_{i=1}^{N}(\hat{\tau}_i^{CT} - Y_i^*)^2$$

   A. Structure and use of criterion
      i. Criterion: $Q^{crit} = Q^{is} - \lambda$ x # leaves
      ii. Select member of set of candidate estimators that maximizes Q^{crit}, given λ
3. Cross-validation approach
   A. Approach: Cross-validation on grid of tuning parameters. Select tuning parameter λ with highest Out-of-sample Goodness-of-Fit Q^{os}.
   B. Out-of-sample Goodness-of-fit function: Q^{os} = -MSE

Causal Trees

- What are you estimating? Within a leaf estimate treatment effect rather than a mean
  - Difference in average outcomes for treated and control group
  - Weight by normalized inverse propensity score in observational studies

- What is your goal? MSE of *treatment effects:* $-E_{S^T}\left[\sum_{i\in S^T}(\tau_i - \hat{\tau}(X_i))^2\right]$

- Problem: this is infeasible (true treatment effect unobserved)
  - We show we can estimate the criteria
- We also modify existing methods to be "honest." We decouple model selection from model estimation.
  - Split sample, one sample to build tree, second to estimate effects.
  - This changes criterion:

$$-E_{S^T,S^E}\left[\sum_{i\in S^T}(\tau_i - \hat{\tau}(X_i; S^E))^2\right]$$

  - Tradeoff:
    - COST: sample splitting means build shallower tree, less personalized predictions, and lower MSE of treatment effects.
    - BENEFIT: Valid confidence intervals with coverage rates that do not deteriorate as data generating process gets more complex or more covariates are added.

Honest Causal Trees

- Honest estimation changes expected criterion
  - Criterion anticipates that we will re-estimate effects in the leaves.
    - The bias due to "dishonest" selection of tree structure will be eliminated.
    - Eliminating the bias was the main purpose of cross-validation in standard method.
  - We face uncertainty in what honest sample will estimate
    - Small leaves will create noise.
    - Splitting on variables that don't affect treatment effect can reduce variance

- Criterion for splitting and cross-validation changes
  - Given set of leaves, MSE on test set taking into account re-estimation.
  - Uncertainty over estimation set and test set at time of evaluation.

Standard *Prediction* Trees

$$\mathrm{MSE}_\mu(\mathcal{S}^{\mathrm{te}}, \mathcal{S}^{\mathrm{est}}, \Pi) \equiv \frac{1}{\#(\mathcal{S}^{\mathrm{te}})} \sum_{i \in \mathcal{S}^{\mathrm{te}}} \left\{ \left(Y_i - \hat{\mu}(X_i; \mathcal{S}^{\mathrm{est}}, \Pi) \right)^2 - Y_i^2 \right\}$$

$$\mathrm{EMSE}_\mu(\Pi) \equiv \mathbb{E}_{\mathcal{S}^{\mathrm{te}}, \mathcal{S}^{\mathrm{est}}} \left[\mathrm{MSE}_\mu(\mathcal{S}^{\mathrm{te}}, \mathcal{S}^{\mathrm{est}}, \Pi) \right]$$

Conventional CART uses for training and CV, respectively:

$$-\mathrm{MSE}_\mu(\mathcal{S}^{\mathrm{tr}}, \mathcal{S}^{\mathrm{tr}}, \Pi) = \frac{1}{N^{\mathrm{tr}}} \sum_{i \in \mathcal{S}^{\mathrm{tr}}} \hat{\mu}^2(X_i; \mathcal{S}^{\mathrm{tr}}, \Pi)$$

$$-\mathrm{MSE}_\mu(\mathcal{S}^{\mathrm{tr,cv}}, \mathcal{S}^{\mathrm{tr,tr}}, \Pi)$$

$$= \frac{1}{N^{tr,cv}} \Sigma_{i \in \mathcal{S}^{tr,cv}} \left((\hat{\mu}(X_i; \mathcal{S}^{tr,tr}))^2 - 2\hat{\mu}(X_i; \mathcal{S}^{tr,cv}) \hat{\mu}(X_i; \mathcal{S}^{tr,tr}) \right)$$

Honest *Prediction* Trees

$$-\text{EMSE}_{\mu}(\Pi) = -\mathbb{E}_{(Y_i,X_i),\mathcal{S}^{\text{est}}}\left[\left(Y_i - \mu(X_i;\Pi)\right)^2 - Y_i^2\right]$$

$$-\mathbb{E}_{X_i,\mathcal{S}^{\text{est}}}\left[\left(\hat{\mu}(X_i;\mathcal{S}^{\text{est}},\Pi) - \mu(X_i;\Pi)\right)^2\right] =$$

$$\mathbb{E}_{X_i}\left[\mu^2(X_i;\Pi)\right] - \mathbb{E}_{\mathcal{S}^{\text{est}},X_i}\left[\mathbb{V}(\hat{\mu}^2(X_i;\mathcal{S}^{\text{est}},\Pi)\right],$$

This uses fact that estimator on independent sample is unbiased

$$\widehat{\mathbb{V}}(\hat{\mu}(x;\mathcal{S}^{\text{est}},\Pi)) \equiv \frac{S^2_{\mathcal{S}^{\text{tr}}}(\ell(x;\Pi))}{N^{\text{est}}(\ell(x;\Pi))}$$

$$\widehat{\mathbb{E}}\left[\mathbb{V}(\hat{\mu}^2(X_i;\mathcal{S}^{\text{est}},\Pi)|i \in \mathcal{S}^{\text{te}}\right] \equiv \frac{1}{N^{\text{est}}} \cdot \sum_{\ell \in \Pi} S^2_{\mathcal{S}^{\text{tr}}}(\ell)$$

$$-\widehat{\text{EMSE}}_{\mu}(\mathcal{S}^{\text{tr}}, N^{\text{est}}, \Pi) \equiv$$

$$\frac{1}{N^{\text{tr}}} \sum_{i \in \mathcal{S}^{\text{tr}}} \hat{\mu}^2(X_i;\mathcal{S}^{\text{tr}},\Pi) - \left(\frac{1}{N^{\text{tr}}} + \frac{1}{N^{\text{est}}}\right) \cdot \sum_{\ell \in \Pi} S^2_{\mathcal{S}^{\text{tr}}}(\ell(x;\Pi))$$

Standard *Causal* Trees

$$\mathrm{MSE}_\tau(\mathcal{S}^{\mathrm{te}}, \mathcal{S}^{\mathrm{est}}, \Pi) \equiv \frac{1}{\#(\mathcal{S}^{\mathrm{te}})} \sum_{i \in \mathcal{S}^{\mathrm{te}}} \left\{ \left(\tau_i - \hat{\tau}(X_i; \mathcal{S}^{\mathrm{est}}, \Pi)\right)^2 - \tau_i^2 \right\}$$

$$\mathrm{EMSE}_\tau(\Pi) \equiv \mathbb{E}_{\mathcal{S}^{\mathrm{te}}, \mathcal{S}^{\mathrm{est}}} \left[\mathrm{MSE}_\tau(\mathcal{S}^{\mathrm{te}}, \mathcal{S}^{\mathrm{est}}, \Pi) \right]$$

$$\widehat{\mathrm{MSE}}_\tau(\mathcal{S}^{\mathrm{te}}, \mathcal{S}^{\mathrm{tr}}, \Pi) \equiv -\frac{2}{N^{\mathrm{tr}}} \sum_{i \in \mathcal{S}^{\mathrm{te}}} \hat{\tau}(X_i; \mathcal{S}^{\mathrm{te}}, \Pi) \cdot \hat{\tau}(X_i; \mathcal{S}^{\mathrm{tr}}, \Pi)$$

$$+\frac{1}{N^{\mathrm{tr}}} \sum_{i \in \mathcal{S}^{\mathrm{te}}} \hat{\tau}^2(X_i; \mathcal{S}^{\mathrm{tr}}, \Pi).$$

For training and CV, respectively:

$$-\widehat{\mathrm{MSE}}_\tau(\mathcal{S}^{\mathrm{tr}}, \mathcal{S}^{\mathrm{tr}}, \Pi) = \frac{1}{N^{\mathrm{tr}}} \sum_{i \in \mathcal{S}^{\mathrm{tr}}} \hat{\tau}^2(X_i; \mathcal{S}^{\mathrm{tr}}, \Pi)$$

$$-\widehat{\mathrm{MSE}}_\tau(\mathcal{S}^{\mathrm{tr,cv}}, \mathcal{S}^{\mathrm{tr,tr}}, \Pi)$$

Honest *Causal* Trees

$$-\text{EMSE}_\tau(\Pi) = \mathbb{E}_{X_i}\left[\tau^2(X_i;\Pi)\right] - \mathbb{E}_{\mathcal{S}^{\text{est}},X_i}\left[\mathbb{V}(\hat{\tau}^2(X_i;\mathcal{S}^{\text{est}},\Pi)\right]$$

This uses fact that estimator on independent sample is unbiased

For training and CV, respectively:

$$-\widehat{\text{EMSE}}_\tau(\mathcal{S}^{\text{tr}},N^{\text{est}},\Pi) \equiv \frac{1}{N^{\text{tr}}}\sum_{i\in\mathcal{S}^{\text{tr}}}\hat{\tau}^2(X_i;\mathcal{S}^{\text{tr}},\Pi)$$

$$-\left(\frac{1}{N^{\text{tr}}}+\frac{1}{N^{\text{est}}}\right)\cdot\sum_{\ell\in\Pi}\left(\frac{S^2_{\mathcal{S}^{\text{tr}}_{\text{treat}}}(\ell)}{p}+\frac{S^2_{\mathcal{S}^{\text{tr}}_{\text{control}}}(\ell)}{1-p}\right)$$

$$-\widehat{\text{EMSE}}_\tau(\mathcal{S}^{\text{tr,cv}},N^{\text{est}},\Pi)$$

Inference

- Attractive feature of trees:
  - Can easily separate tree construction from treatment effect estimation
  - Tree constructed on training sample is indep. of sampling variation in test sample
  - Holding tree from training sample fixed, can use standard methods to conduct inference within each leaf of the tree on test sample
    - Use any valid method for treatment effect estimation, not just method used in training. Asymptotic theory as usual *within a leaf*.
  - Once you have the partition, just run a regression on second sample interacting leaf dummies with treatment indicator. Everything is as usual.

- We do not require ANY assumptions about sparsity of true data-generating process. *Coverage* does not deteriorate at all as you increase number of covariates.
  - But we do not attempt to make fully personalized estimates

Comparing Alternative Approaches to Preferred Honest Causal Tree

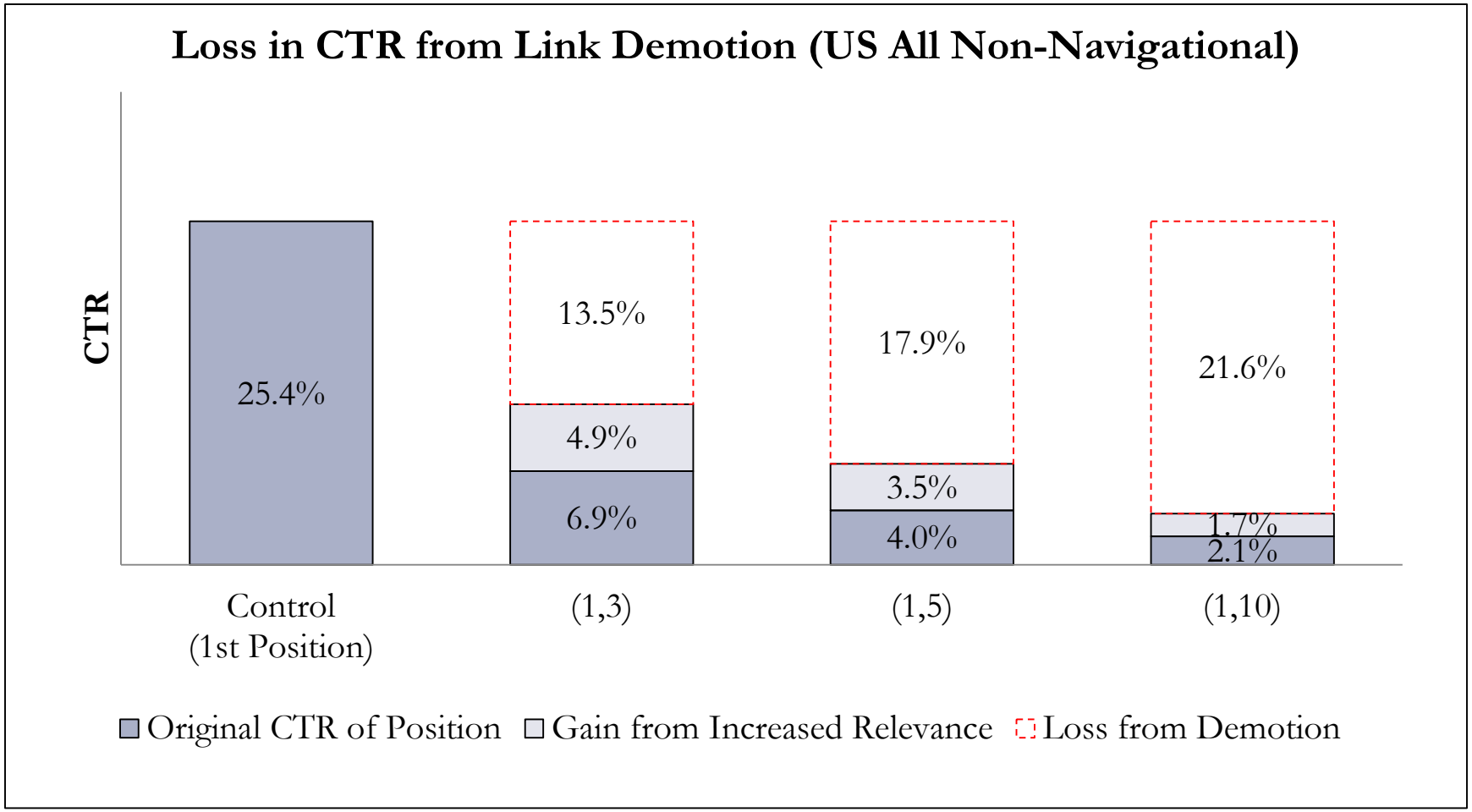
- Dishonest with double the sample
  - Does worse if true model is sparse (also the case where bias is less severe)
  - Has similar or better MSE in many cases, but poor coverage of confidence intervals

- Splitting on statistical criteria of model fit
  - Paper shows formally how these methods differ (proposed in a small related literature, one that doesn't consider honesty and cross-validation issues)
  - Splitting on T-statistic on treatment effect ignores variance reduction from reducing imbalance on covariates
  - Splitting on overall model fit prioritizes level heterogeneity above treatment effects

Application: Treatment Effect Heterogeneity in Estimating Position Effects in Search

- Queries highly heterogeneous
  - Tens of millions of unique search phrases each month
  - Query mix changes month to month for a variety of reasons
  - Behavior conditional on query is fairly stable
- Desire for segments.
  - Want to understand heterogeneity and make decisions based on it
  - "Tune" algorithms separately by segment
  - Want to predict outcomes if query mix changes
    - For example, bring on new syndication partner with more queries of a certain type

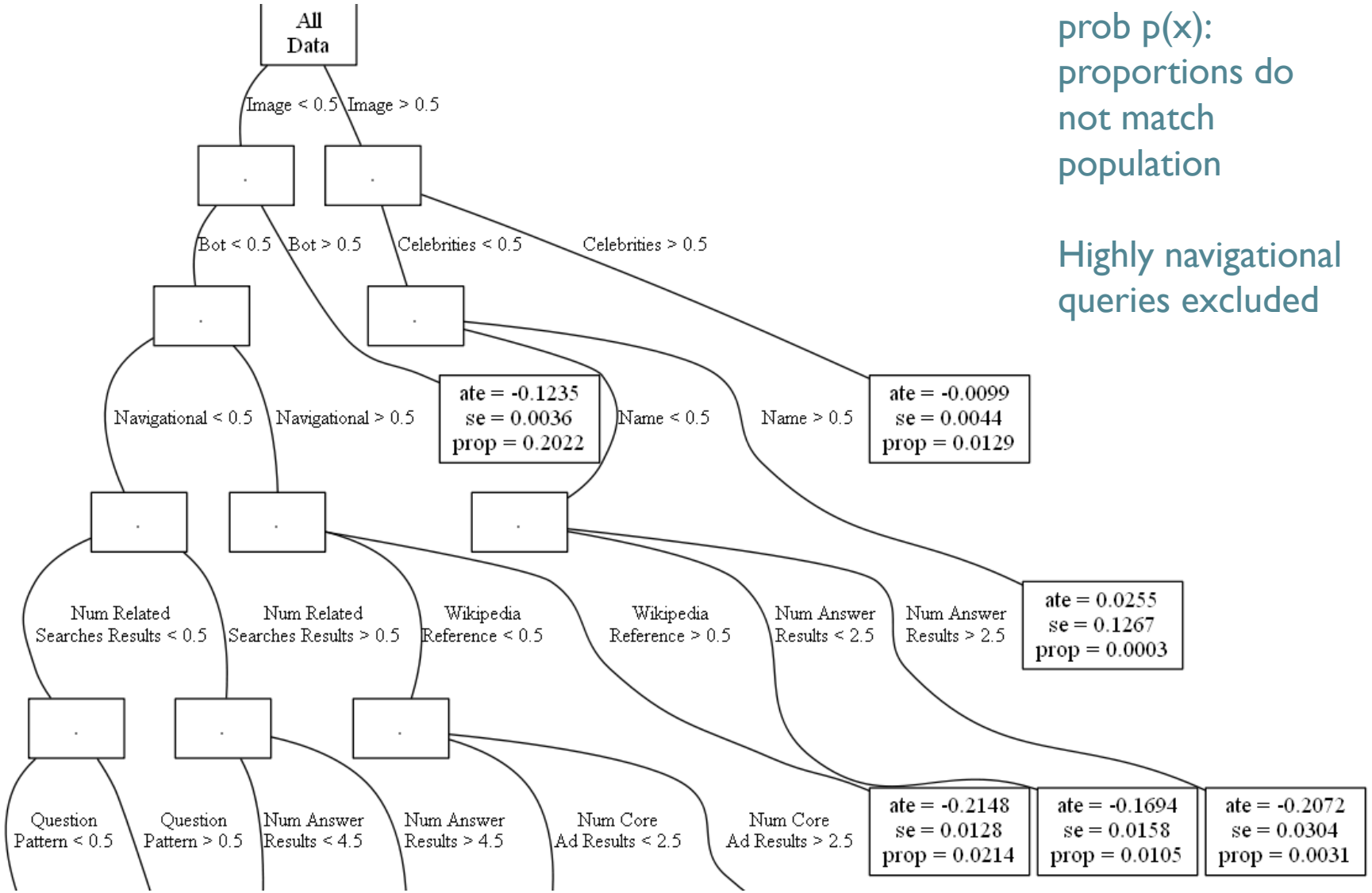
Relevance v. Position

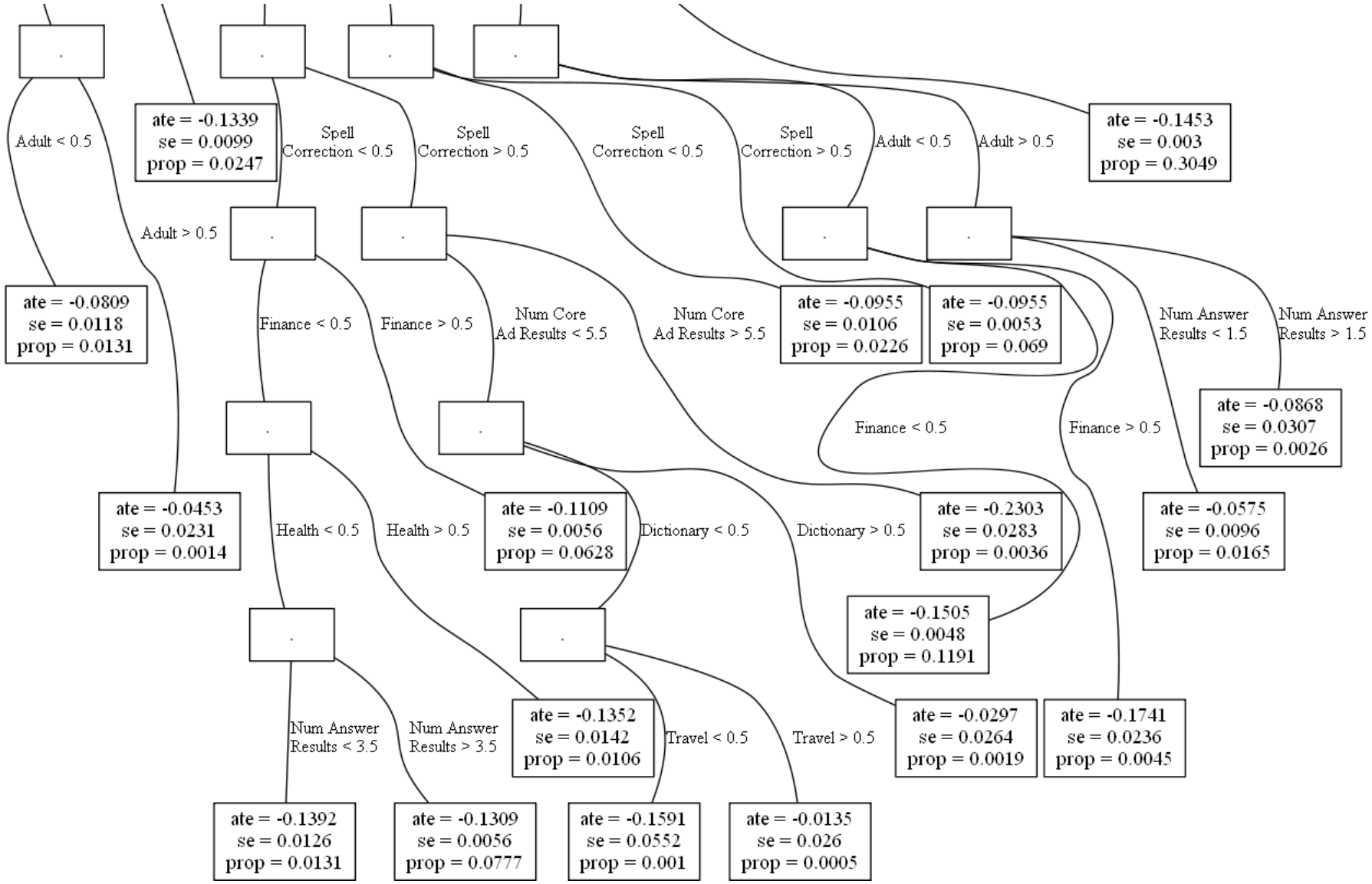


Search Experiment Tree: Effect of Demoting Top Link (Test Sample Effects)

Some data excluded with prob p(x): proportions do not match population

Highly navigational queries excluded

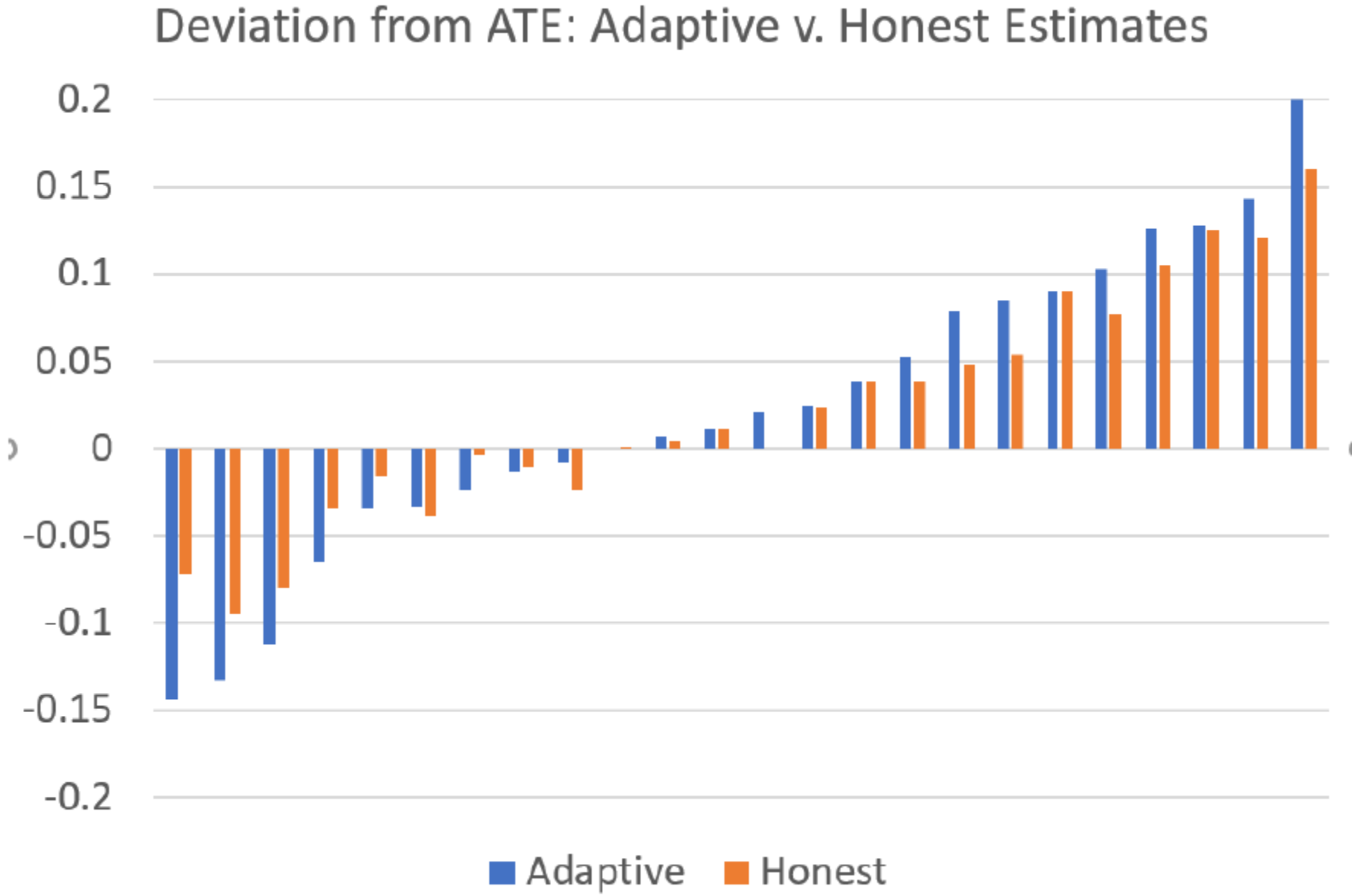



Adult < 0.5
ate = -0.1339
se = 0.0099
prop = 0.0247
Spell Correction < 0.5
Spell Correction > 0.5
Spell Correction < 0.5
Spell Correction > 0.5
Adult < 0.5
Adult > 0.5
ate = -0.1453
se = 0.003
prop = 0.3049
Adult > 0.5
ate = -0.0809
se = 0.0118
prop = 0.0131
Finance < 0.5
Finance > 0.5
Num Core Ad Results < 5.5
Num Core Ad Results > 5.5
ate = -0.0955
se = 0.0106
prop = 0.0226
ate = -0.0955
se = 0.0053
prop = 0.069
Num Answer Results < 1.5
Num Answer Results > 1.5
ate = -0.0868
se = 0.0307
prop = 0.0026
Finance < 0.5
Finance > 0.5
ate = -0.0453
se = 0.0231
prop = 0.0014
Health < 0.5
Health > 0.5
ate = -0.1109
se = 0.0056
prop = 0.0628
Dictionary < 0.5
Dictionary > 0.5
ate = -0.2303
se = 0.0283
prop = 0.0036
ate = -0.0575
se = 0.0096
prop = 0.0165
ate = -0.1505
se = 0.0048
prop = 0.1191
Num Answer Results < 3.5
Num Answer Results > 3.5
ate = -0.1352
se = 0.0142
prop = 0.0106
Travel < 0.5
Travel > 0.5
ate = -0.0297
se = 0.0264
prop = 0.0019
ate = -0.1741
se = 0.0236
prop = 0.0045
ate = -0.1392
se = 0.0126
prop = 0.0131
ate = -0.1309
se = 0.0056
prop = 0.0777
ate = -0.1591
se = 0.0552
prop = 0.001
ate = -0.0135
se = 0.026
prop = 0.0005

Use Test Sample for Segment Means & Std Errors to Avoid Bias

Variance of estimated treatment effects in training sample 2.5 times that in test sample (adaptive estimates biased)

Honest Estimates			Adaptive Estimates		
Treatment Effect	Standard Error	Proportion	Treatment Effect	Standard Error	Proportion
-0.124	0.004	0.202	-0.124	0.004	0.202
-0.134	0.010	0.025	-0.135	0.010	0.024
-0.010	0.004	0.013	-0.007	0.004	0.013
-0.215	0.013	0.021	-0.247	0.013	0.022
-0.145	0.003	0.305	-0.148	0.003	0.304
-0.111	0.006	0.063	-0.110	0.006	0.064
-0.230	0.028	0.004	-0.268	0.028	0.004
-0.058	0.010	0.017	-0.032	0.010	0.017
-0.087	0.031	0.003	-0.056	0.029	0.003
-0.151	0.005	0.119	-0.169	0.005	0.119
-0.174	0.024	0.005	-0.168	0.024	0.005
0.026	0.127	0.000	0.286	0.124	0.000
-0.030	0.026	0.002	-0.009	0.025	0.002
-0.135	0.014	0.011	-0.114	0.015	0.010
-0.159	0.055	0.001	-0.143	0.053	0.001
-0.014	0.026	0.001	0.008	0.050	0.000
-0.081	0.012	0.013	-0.050	0.012	0.013
-0.045	0.023	0.001	-0.045	0.021	0.001
-0.169	0.016	0.011	-0.200	0.016	0.011
-0.207	0.030	0.003	-0.279	0.031	0.003
-0.096	0.011	0.023	-0.083	0.011	0.022
-0.096	0.005	0.069	-0.096	0.005	0.070
-0.139	0.013	0.013	-0.159	0.013	0.013
-0.131	0.006	0.078	-0.128	0.006	0.078

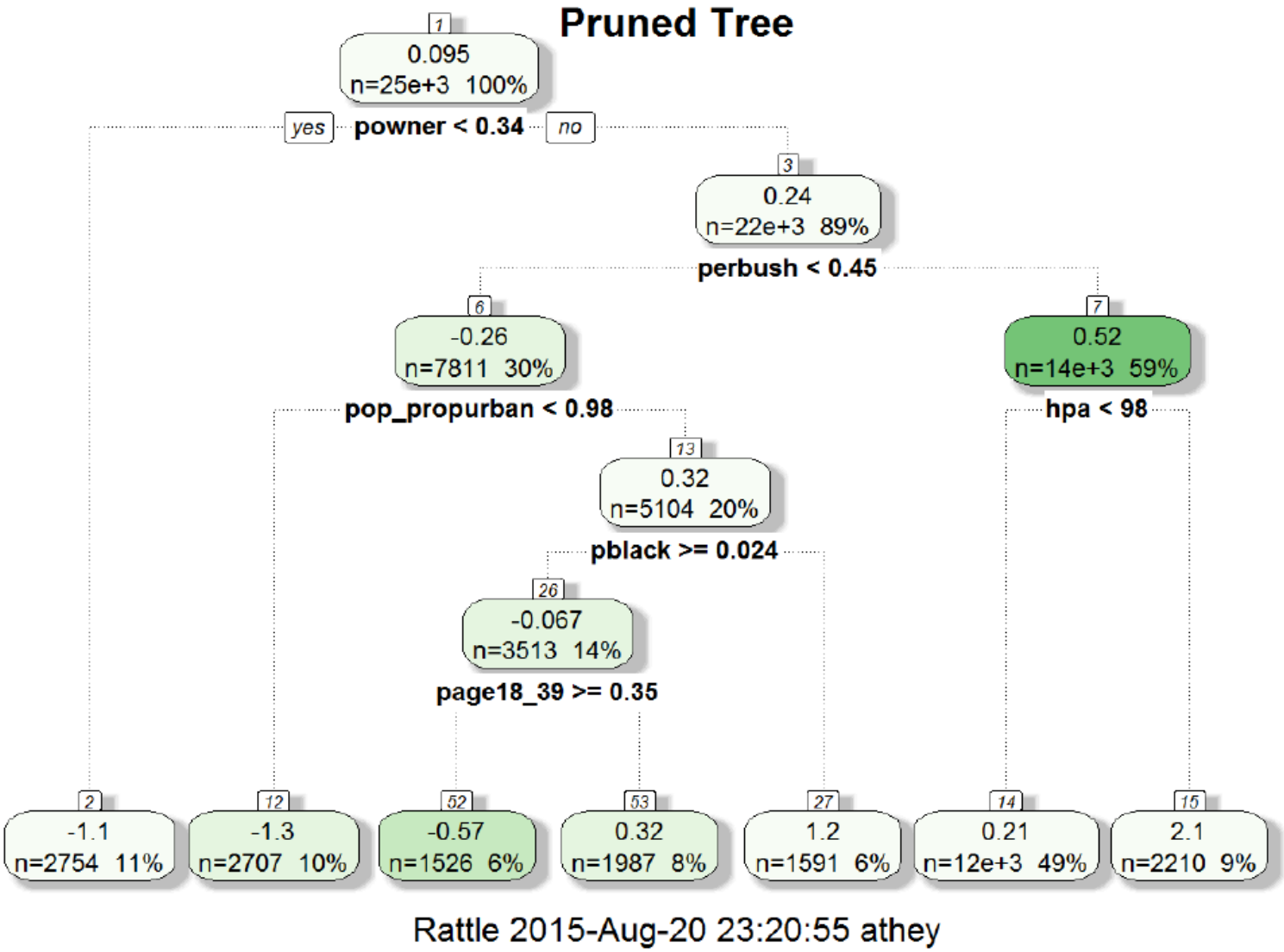

Deviation from ATE: Adaptive v. Honest Estimates
0.2
0.15
0.1
0.05
0
-0.05
-0.1
-0.15
-0.2
Adaptive
Honest

Analyzing Field Experiments: Revisit Karlan and List (AER)

- Field experiment on charitable giving
  - Solicitations: treatment groups are offered match of various sizes, example gifts, and limits
  - Finding: match increases gift amount, but larger sizes do not have incremental effect
  - Some exploration of heterogeneity

- Apply causal trees:
  - Explore heterogeneity more systematically
  - Highlight the risks of data mining
  - Unlike the search application, it appears there isn't a lot of treatment effect heterogeneity

Effect of All Treatments (Pooled) Training Data



Training Sample

```
Coefficients:
                           Estimate Std. Error t value Pr(>|t|)
leavesr-1.342               1.9395     0.2768   7.006 2.51e-12 ***
leavesr-1.083               1.6189     0.2845   5.691 1.28e-08 ***
leavesr-0.573               0.9498     0.3696   2.570  0.01018 *
leavesr0.213                0.3260     0.1350   2.415  0.01574 *
leavesr0.321                0.6566     0.3257   2.016  0.04379 *
leavesr1.171                0.2839     0.3669   0.774  0.43907
leavesr2.086                1.6176     0.3288   4.921 8.68e-07 ***
leavesr-1.342:treatment    -1.3417     0.3445  -3.895 9.86e-05 ***
leavesr-1.083:treatment    -1.0826     0.3475  -3.116  0.00184 **
leavesr-0.573:treatment    -0.5733     0.4593  -1.248  0.21201
leavesr0.213:treatment      0.2131     0.1648   1.294  0.19581
leavesr0.321:treatment      0.3210     0.4035   0.796  0.42632
leavesr1.171:treatment      1.1707     0.4527   2.586  0.00972 **
leavesr2.086:treatment      2.0856     0.3951   5.279 1.31e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Estimation Sample

Coefficients:

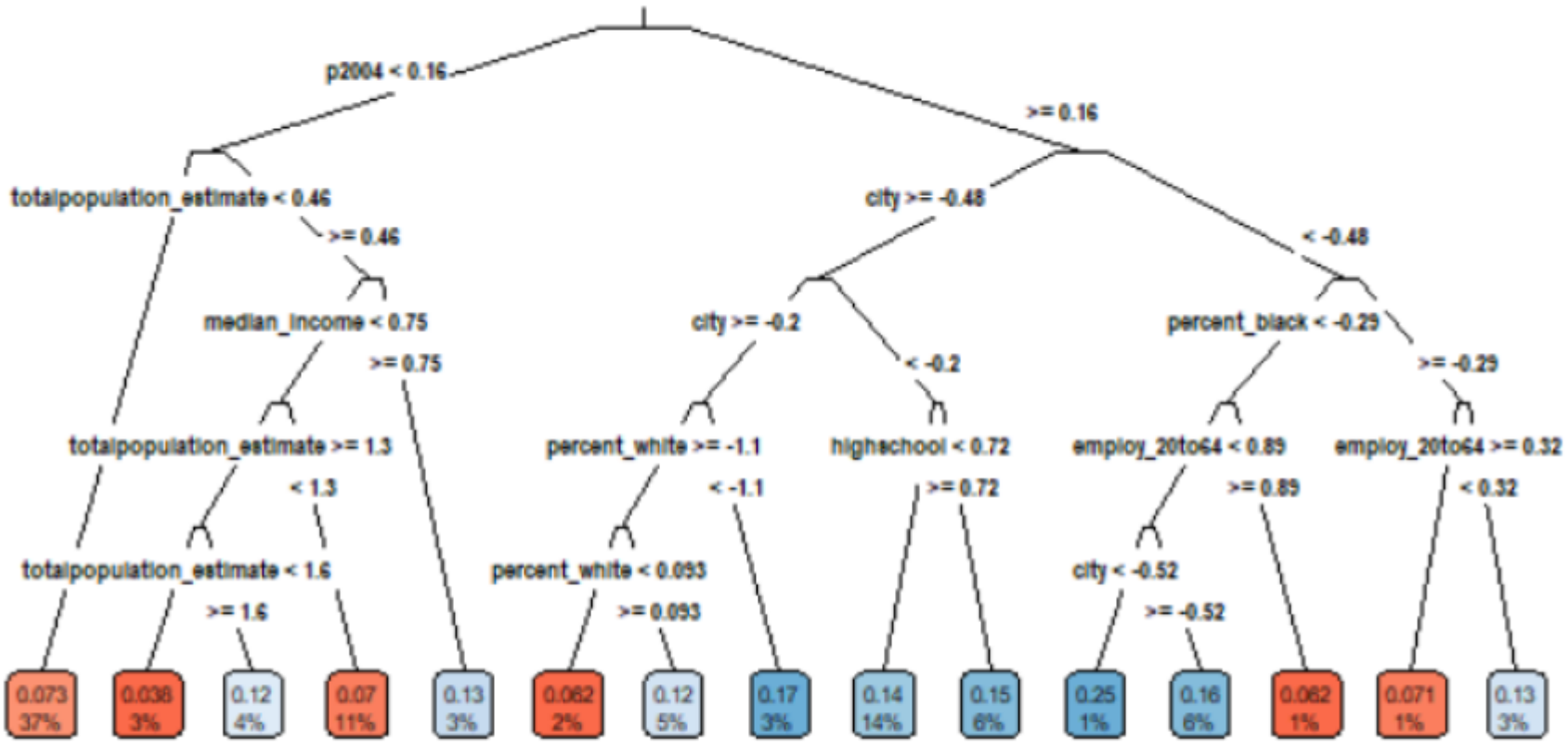
	Estimate	Std. Error	t value	Pr(>|t|)	
leavesr-1.342	0.62900	0.28579	2.201	0.02775	*
leavesr-1.083	0.51463	0.28810	1.786	0.07407	.
leavesr-0.573	0.75687	0.40245	1.881	0.06003	.
leavesr0.213	0.34395	0.13808	2.491	0.01275	*
leavesr0.321	1.07704	0.34707	3.103	0.00192	**
leavesr1.171	1.09966	0.36281	3.031	0.00244	**
leavesr2.086	3.39262	0.31770	10.679	< 2e-16	***
leavesr-1.342:treatment	0.37260	0.34983	1.065	0.28685	
leavesr-1.083:treatment	0.29186	0.35475	0.823	0.41067	
leavesr-0.573:treatment	0.10772	0.48868	0.220	0.82553	
leavesr0.213:treatment	0.17869	0.16888	1.058	0.29002	
leavesr0.321:treatment	-0.28147	0.42071	-0.669	0.50347	
leavesr1.171:treatment	-0.08332	0.44850	-0.186	0.85262	
leavesr2.086:treatment	0.89775	0.39055	2.299	0.02153	*

Interpretation: Sample Splitting is Key

- Training set makes it look like there is lots of heterogeneity
- Estimation sample accurately shows most of that is spurious
- You get the right, if disappointing, answer out of the estimation sample

Voter Turnout Example

- Gerber, Green, and Larimer (2008)s paper Social Pressure and Voter Turnout: Evidence from a Large-Scale Field Experiment
  - Look at mailing that tells people that their neighbors will be informed about whether they voted
  - Look for heterogeneous treatment effects using Causal Tree

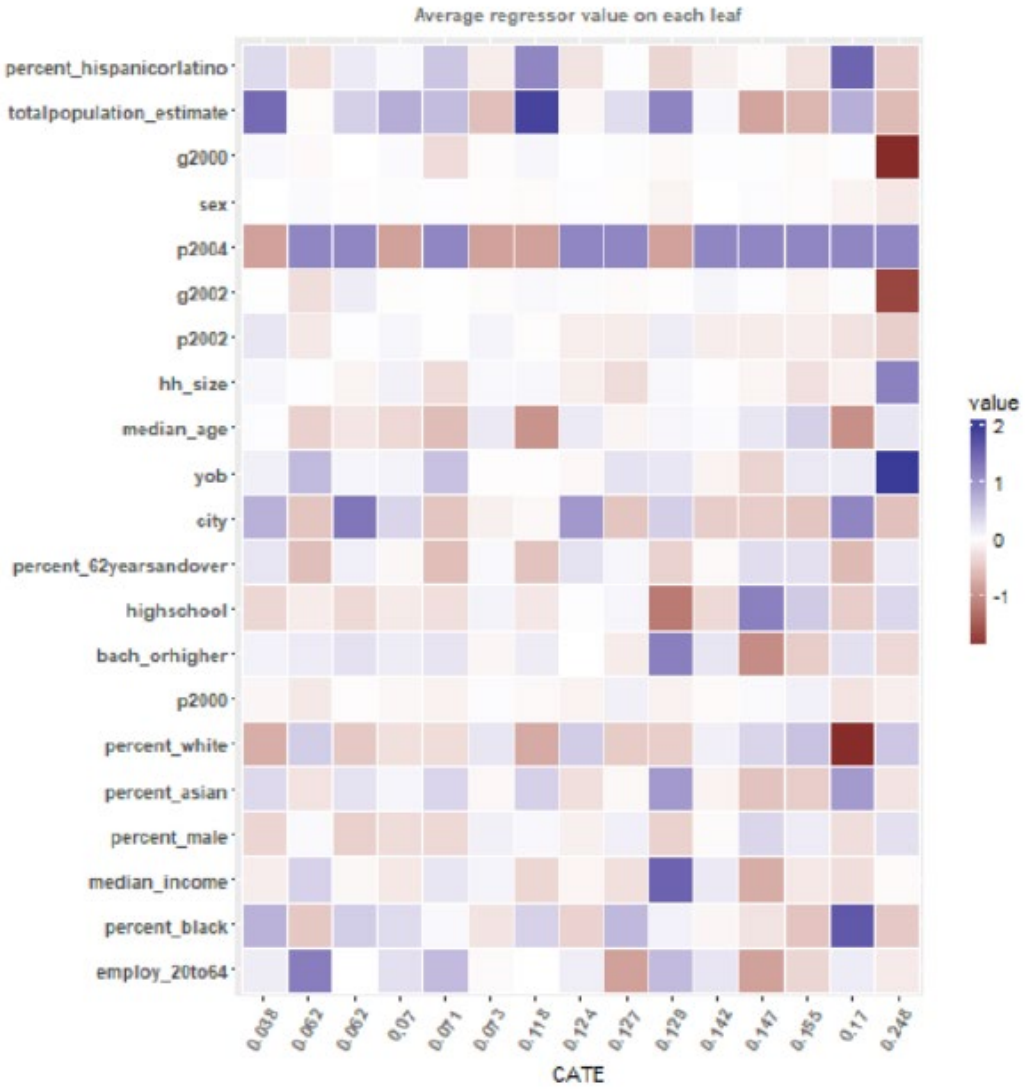


Note: in this application, trees are not very stable, and improvement over null model is negligible. Might need more aggressive pruning criterion.

Describing Results

- WRONG: The tree didn't split on voting in 2002, so that is not important"
  - The mistake is that there are multiple ways to describe almost identical subgroups. The tree identified subgroups one way, but there could have been many others with equal explanatory power
- RIGHT: These are the characteristics of the subgroups
  - In practice: Describe the characteristics and test differences between them

Describing Characteristics of Leaves



Testing hypotheses about heterogeneity across leaves

	0.038	0.0619	0.0623	0.0702	0.0712	0.0734	0.1181	0.1235	0.1275	0.1294	0.1421	0.1467	0.1551	0.1703	0.2481
0.038															
0.0619	0.637														
0.0623	0.637	0.993													
0.0702	0.402	0.836	0.849												
0.0712	0.512	0.857	0.866	0.979											
0.0734	0.325	0.760	0.777	0.860	0.954										
0.1181	0.066	0.212	0.226	0.118	0.299	0.105									
0.1235	0.044	0.160	0.173	0.065	0.234	0.051	0.879								
0.1275	0.038	0.141	0.153	0.055	0.207	0.043	0.797	0.910							
0.1294	0.037	0.135	0.147	0.055	0.199	0.044	0.762	0.870	0.958						
0.1421	0.010	0.056	0.064	0.005	0.092	0.002	0.470	0.556	0.652	0.703					
0.1467	0.021	0.080	0.088	0.031	0.120	0.026	0.487	0.561	0.636	0.675	0.903				
0.1551	0.003	0.025	0.030	0.001	0.045	0.000	0.260	0.311	0.389	0.435	0.646	0.821			
0.1703	0.004	0.023	0.027	0.004	0.039	0.002	0.196	0.231	0.282	0.312	0.444	0.592	0.678		
0.2481	0.077	0.119	0.121	0.121	0.139	0.126	0.266	0.284	0.301	0.309	0.359	0.390	0.421	0.509	

Note: p-values uncorrected for multiple testing

Summary

- Key to approach
  - Distinguish between causal and predictive parts of model

- Combining two literatures
  - Combining very well established tools from different literatures
  - Systematic model selection with many covariates
  - Optimized for problem of causal effects
    - In terms of tradeoff between granular prediction and overfitting
  - With valid inference
    - While sacrificing fully personalized predictions, but gaining…
  - Easy to communicate method and interpret results
    - Output is a partition of sample, treatment effects and standard errors