Efficiently Exploring Multilevel Data with Recursive Partitioning

Daniel P. Martin
University of Virginia

May 26, 2015

Outline

- Exploratory data analysis discussion
- Intro to recursive partitioning
- Multilevel extensions
- Multilevel issues (and "best practices")

The term "exploratory" is considered by many as less than an approach to data analysis and more a confession of guilt.

-Jack McArdle, 2014

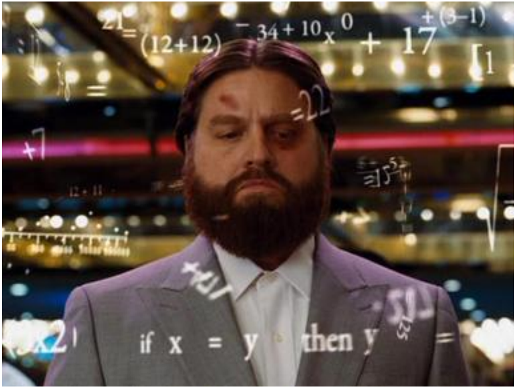
The term "exploratory" is considered by many as less than an approach to data analysis and more a confession of guilt.

-Jack McArdle, 2014

Why is exploratory research seen in this way? How do you use exploratory research (if at all)?

What Typically Comes After Confirmatory Tests?

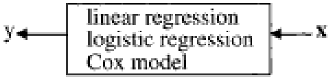
Data-driven exploration with NHST



If Not NHST, What Else Is There?

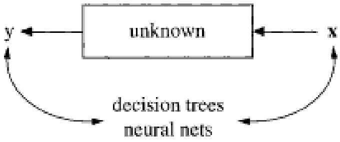
Breiman (2001)

Data modeling:

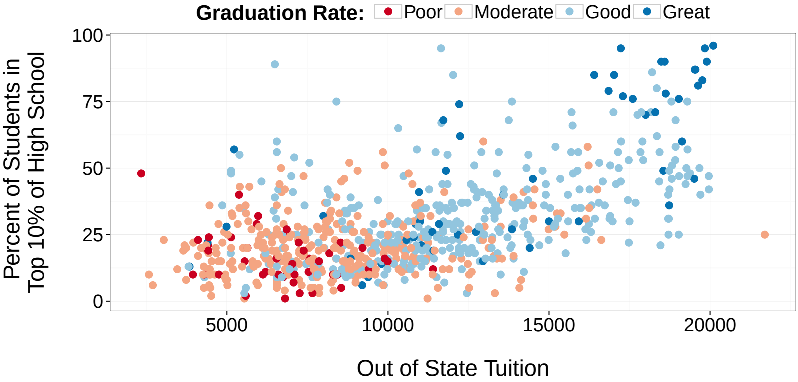


vs.

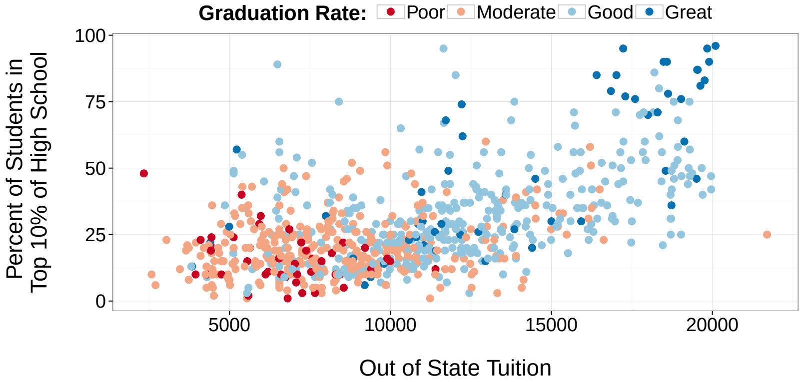
Algorithmic modeling:



An Introduction to Decision Trees

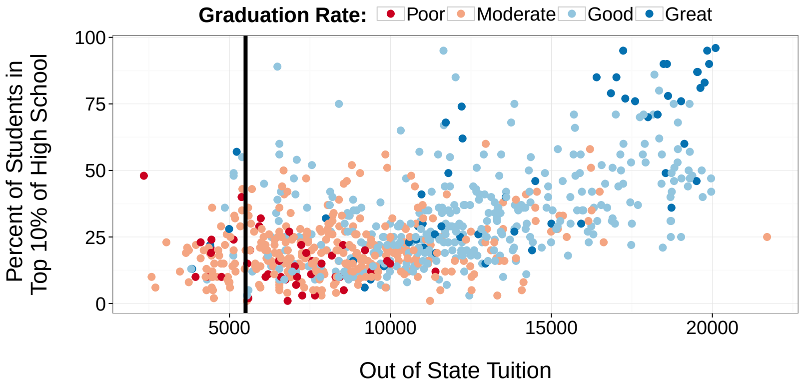


An Introduction to Decision Trees



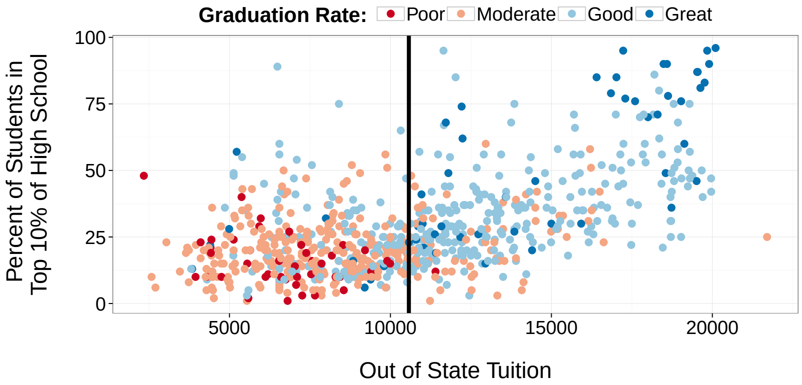
$$RSS_{total} \overset{?}{<} RSS_{part1} + RSS_{part2}$$

An Introduction to Decision Trees



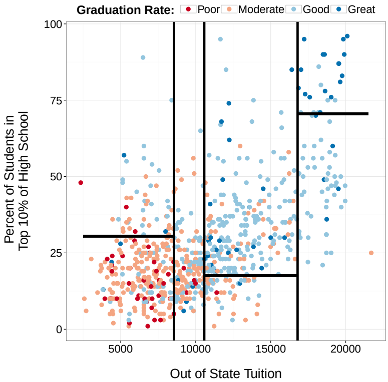
$$229 \overset{?}{<} 19 + 195 = 215$$

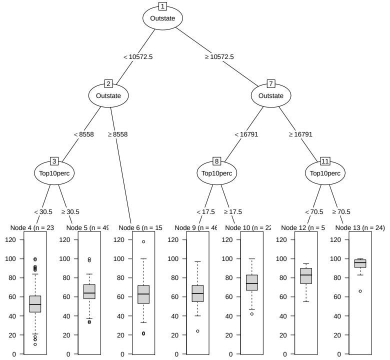
An Introduction to Decision Trees



$$229 \overset{?}{<} 104 + 63 = 168$$

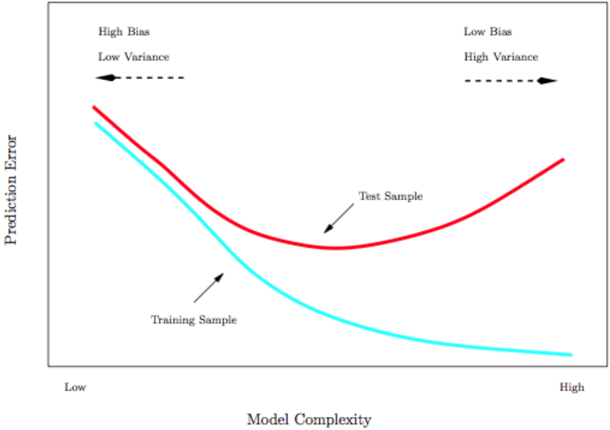
An Introduction to Decision Trees





Detour: The Bias-Variance Tradeoff

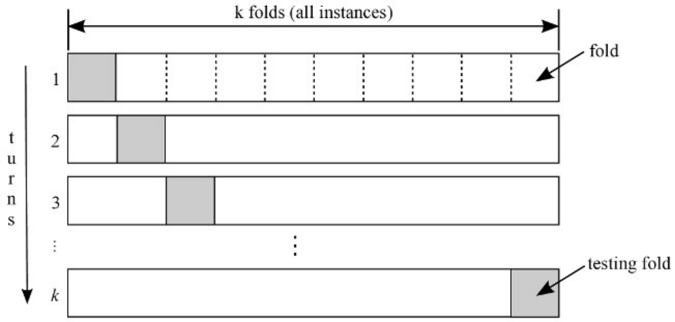
training, testing, and cross-validation



Source: *An Introduction to Statistical Learning*

Detour: The Bias-Variance Tradeoff

training, testing, and cross-validation



Decision Tree Pseudocode

CART: Breiman et al. (1984)

1. Search all variables for splits in a greedy, top-down manner

Decision Tree Pseudocode

CART: Breiman et al. (1984)

1. Search all variables for splits in a greedy, top-down manner
2. Identify the best split by some criterion

Decision Tree Pseudocode

CART: Breiman et al. (1984)

1. Search all variables for splits in a greedy, top-down manner
2. Identify the best split by some criterion
3. Split the sample on this threshold, resulting in two child nodes

Decision Tree Pseudocode

CART: Breiman et al. (1984)

1. Search all variables for splits in a greedy, top-down manner
2. Identify the best split by some criterion
3. Split the sample on this threshold, resulting in two child nodes
4. Repeat steps 1-3 on the resulting nodes until some stopping criterion is reached

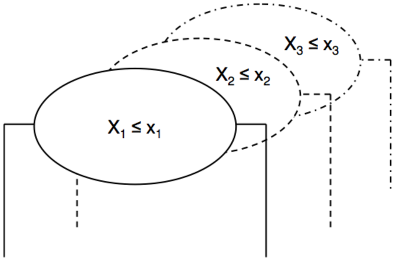
Decision Tree Pseudocode

CART: Breiman et al. (1984)

1. Search all variables for splits in a greedy, top-down manner
2. Identify the best split by some criterion
3. Split the sample on this threshold, resulting in two child nodes
4. Repeat steps 1-3 on the resulting nodes until some stopping criterion is reached
5. Prune tree using cross-validation

Handling Missingness - Decision Trees

surrogate splits



Source: *Hapfelmeier (2012)*

Recap: Pros and Cons of Decision Trees

Pros:

- intuitive, easy to explain and visualize
- can handle continuous or categorical outcomes
- non-parametric, robust to outliers
- no model specification required

Recap: Pros and Cons of Decision Trees

Pros:

- intuitive, easy to explain and visualize
- can handle continuous or categorical outcomes
- non-parametric, robust to outliers
- no model specification required

Cons:

- biased toward variables with many possible splits
- typically outperformed by regression techniques
- prone to overfitting

Random Forest Pseudocode

CART forests: Breiman (2001)

1. Take a bootstrap sample

Random Forest Pseudocode

CART forests: Breiman (2001)

1. Take a bootstrap sample
2. Select a random subset of predictors

Random Forest Pseudocode

CART forests: Breiman (2001)

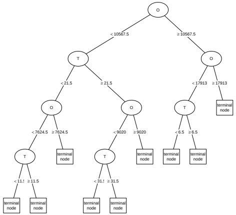
1. Take a bootstrap sample
2. Select a random subset of predictors
3. Fit a decision tree to full depth

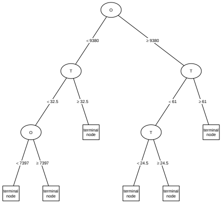
Random Forest Pseudocode

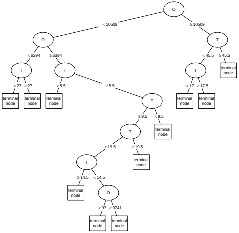
CART forests: Breiman (2001)

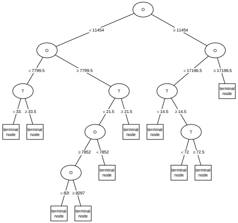
1. Take a bootstrap sample
2. Select a random subset of predictors
3. Fit a decision tree to full depth
4. Repeat 500ish times

Creating Ensembles of Trees



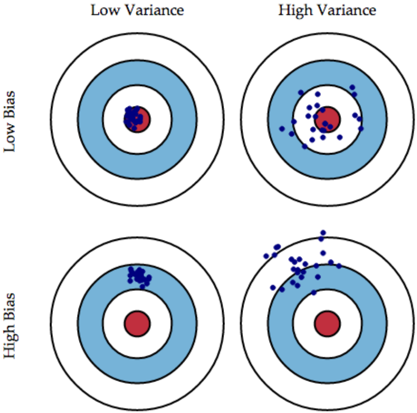






Creating Ensembles of Trees

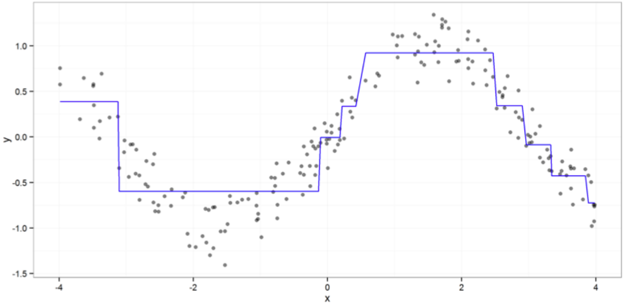
why it works - theoretical



Source: *Scott Fortmann-Roe*

Creating Ensembles of Trees

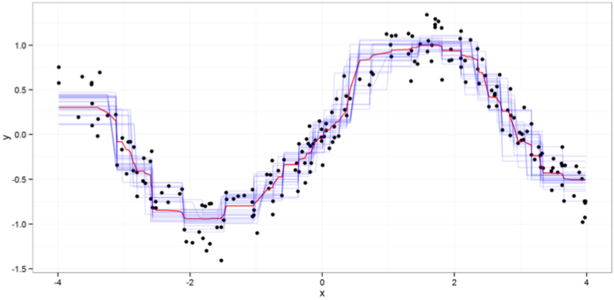
why it works - applied



Source: *Zachary Jones*

Creating Ensembles of Trees

why it works - applied



Source: *Zachary Jones*

Handling Missingness - Forests

imputation by proximity

For missing data:

1. Calculate proximity matrix (number of times observations show up in the same node)
2. Impute missing values using medians and levels of the highest frequency
3. Run a random forest model
4. Update missing values to a weighted mean of the observations or category with the largest average proximity
5. Repeat 5-10 times

Recap: Pros and Cons of Decision Trees

Pros:

- All the CART pros!
- Can now approximate smooth, nonlinear relationships instead of piecewise constant fits
- Unlikely to overfit
- Not much tuning required compared to other algorithmic methods

Recap: Pros and Cons of Decision Trees

Pros:

- All the CART pros!
- Can now approximate smooth, nonlinear relationships instead of piecewise constant fits
- Unlikely to overfit
- Not much tuning required compared to other algorithmic methods

Cons:

- still biased toward variables with many possible splits
- Harder to interpret
- Longer computation time (still manageable for large datasets)

Interpreting the Black Box

1. Variable Importance
2. Partial Dependence Plots

Interpreting the Black Box

1. Variable Importance
2. Partial Dependence Plots

more on this in a sec...

Not a "Magic" Solution

Not a “Magic” Solution

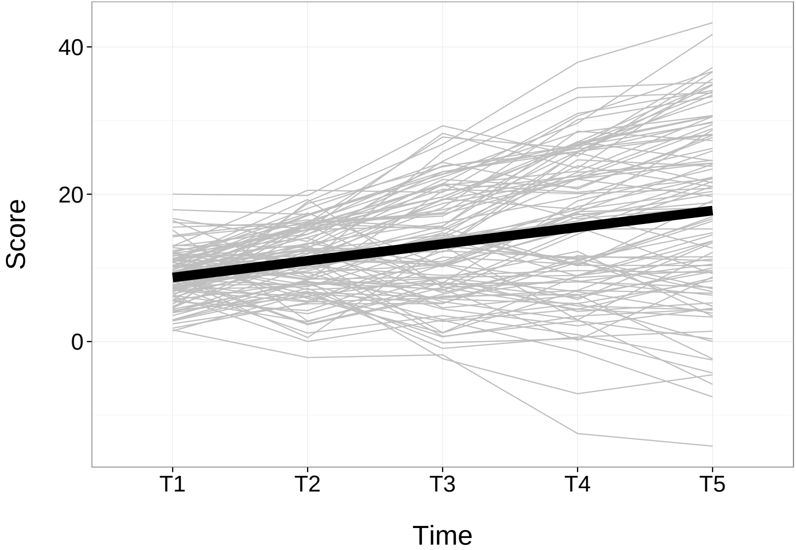
Random forests make no general assumptions regarding independence, and thus have the potential to be used for multilevel EDA with **little added complexity**

Not a “Magic” Solution

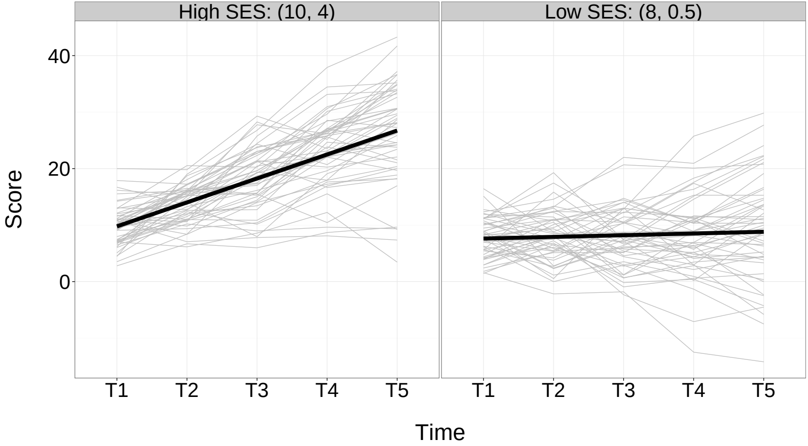
Random forests make no general assumptions regarding independence, and thus have the potential to be used for multilevel EDA with **little added complexity**

However, not much is known about what happens when forests are used in this way

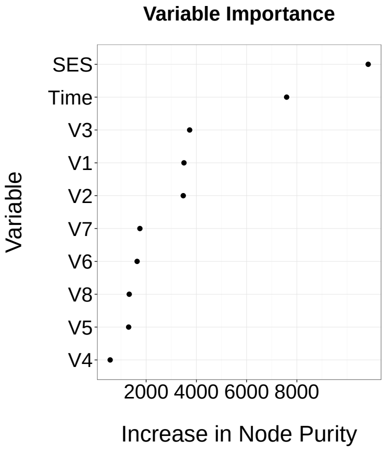
Proof of Concept

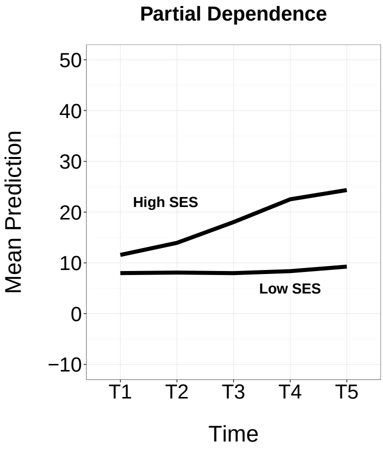


Proof of Concept



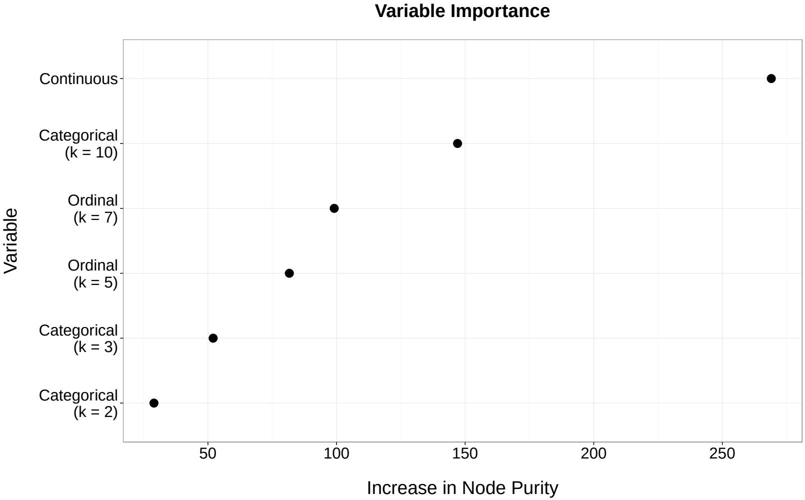
Proof of Concept





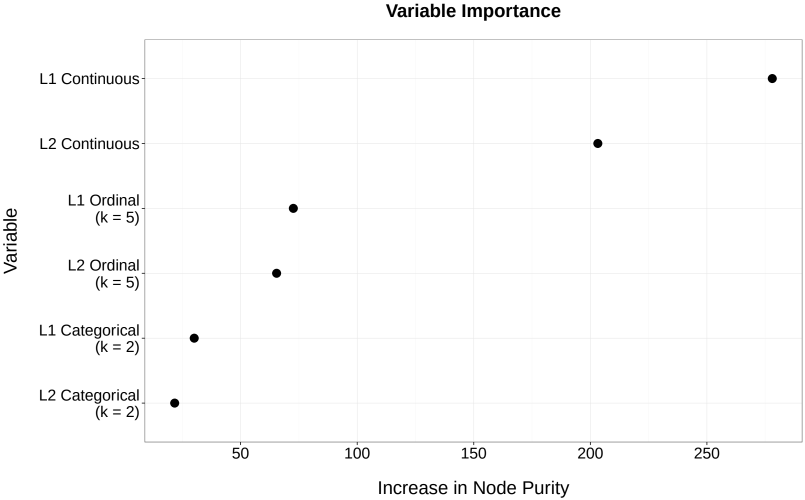
Issue 1: CART biased variable selection

single level (N = 1000)



Issue 1: CART biased variable selection

multilevel (N = 1000, L2/L1 = 100/10)



Issue 2: Underestimation of OOB error

$$P_{notselected} = (1 - \frac{1}{n})^n$$

$$\lim_{n \to \infty} P = \frac{1}{e} \approx 0.368$$

Issue 2: Underestimation of OOB error

$$MSE_{test} = 48.32$$

$$MSE_{OOB} = 23.95$$

Reminder: Issues to Keep in Mind

- OOB errer estimates will be unreliable
- Additional bias for level-2 variables occurs
- DO NOT use this method and then perform confirmatory tests on the same data

Analysis Steps

1. Initial pre-processing ("feature engineering", handle missingness)
2. Estimate ICC and consider what level the variables were measured at
3. Estimate predictive performance using a hold out test set or cross-validation (at level-2)
4. Examine variable importance and partial dependence plots

Helpful (and Accessible) Citations

Breiman, L. (2001). Statistical modeling: The two cultures
Shmueli, G. (2010). To explain or predict?
Strobl, C. et al. (2009). An introduction to recursive partitioning: rationale, application, and characteristics of classification and regression trees, bagging, and random forests