Introduction to Decision Trees and Their Applications in Microbial Ecology

Michelle Berry

DANG February 2015

Overview

1) Introduce the concept of Statistical Learning

2) Decision trees

3) Random forests

4) Resources

Some of the figures in this presentation are taken from:
"An Introduction to Statistical Learning, with applications in R"
(Springer, 2013) with permission from the authors: G. James, D. Witten, T. Hastie and R. Tibshirani

What is Statistical Learning?

- A set of algorithms for understanding and visualizing high-dimensional data. Most models have an emphasis on Prediction.
- The algorithm Learns from the data.
- Traditional modeling is Top-Down, statistical learning is Bottom-Up.
- Statistical learning models are sometimes less Interpretable.

How does the model learn?

- Split data into two parts: Training Data and Test Data.
- Use the Training Data to fit the model's parameters.
- Use the Test Data to validate the model or measure its prediction accuracy.

A Quick Definition

Response/
outcome variable

Predictor
variables

$$Y = B_1X_1 + B_2X_2 + B_3X_3$$

What Are Tree Methods ?

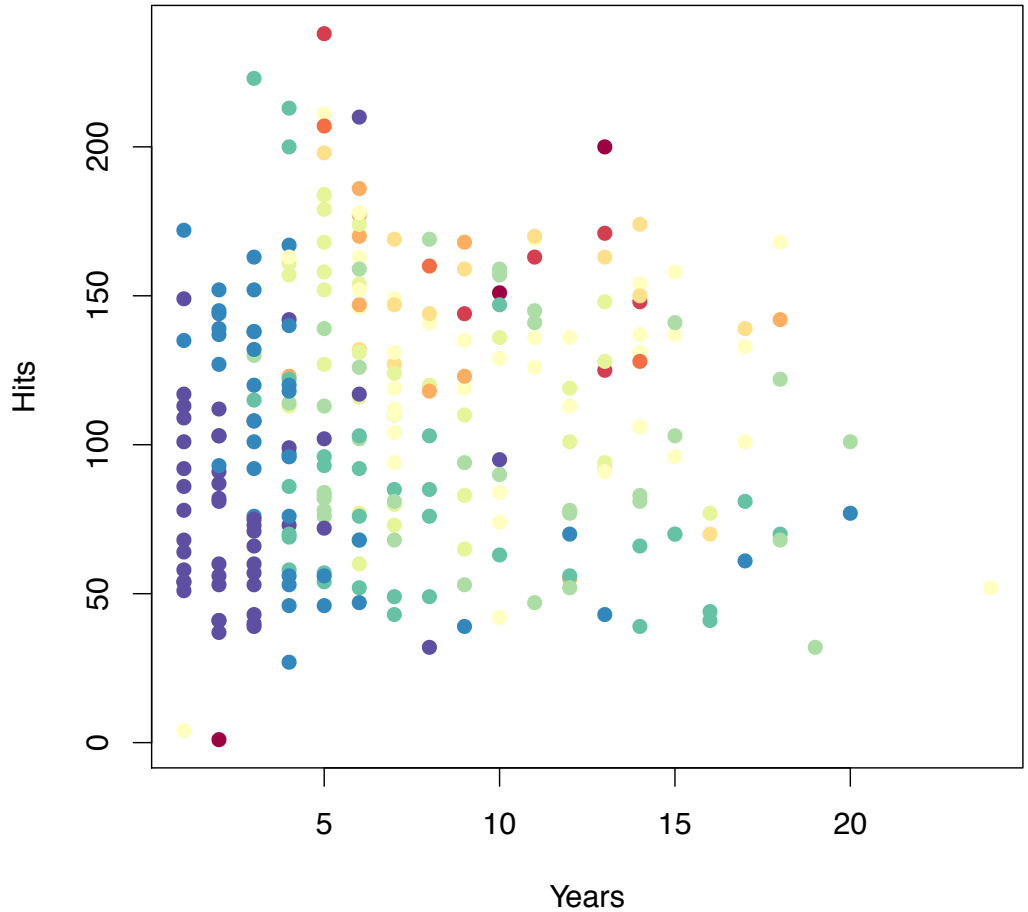
- The results can be visualized in a Tree
- They involve segmenting the Predictor Space into a number of simple regions.
- Tree methods can be used for both:
  - Regression (continuous response variable)
  - Classification (discrete response variable)

Tree Algorithm

- Divide the predictor space – the set of possible values for $X_1, X_2, \ldots X_p$ – into J distinct and non-overlapping regions, $R_1, R_2, \ldots R_j$
- At each split, we select the predictor X_j and the cutpoint that leads to the greatest reduction in error rate (classification error or regression residuals)
- For every observation that falls into the region Rj, we make the same prediction, which is simply the mean of the response values for all training observations in R_j

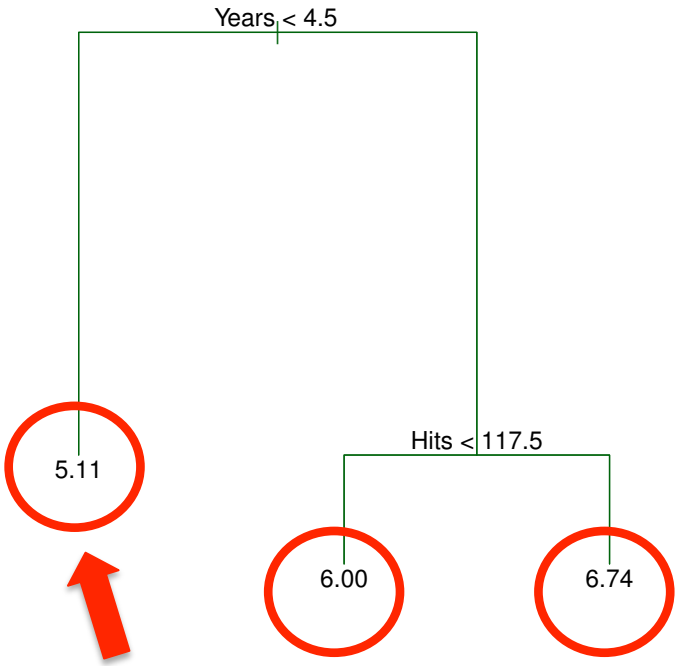
Example: regression tree

How much $ does a pro baseball player make?

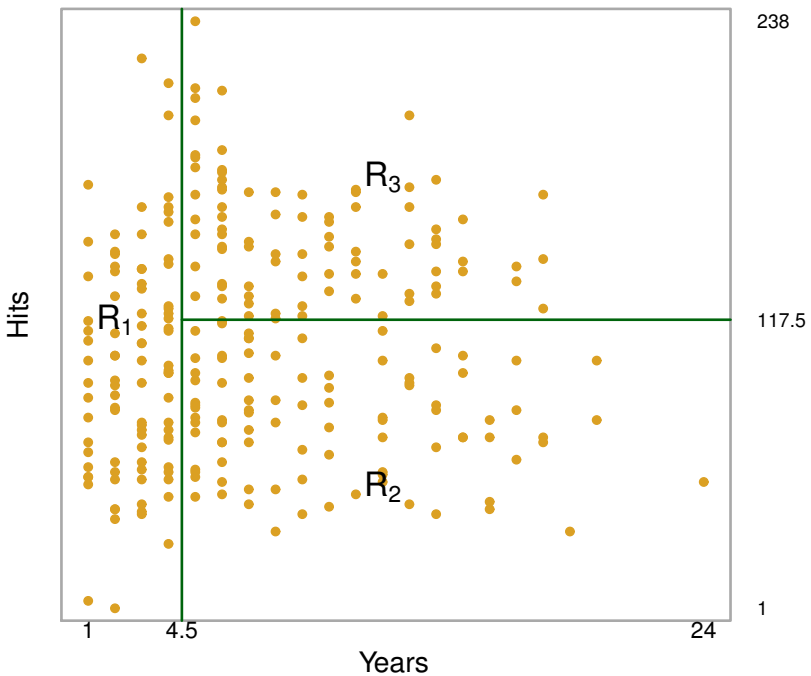


blue/green = Low salary, yellow/red = high salary

Example: regression tree



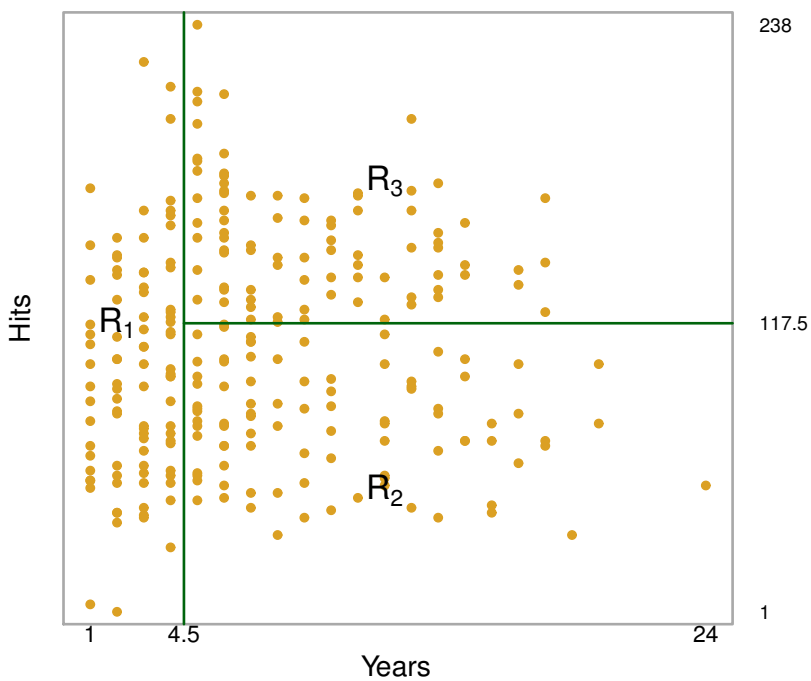
Predictor Space:



Regression trees have
Continuous response variables

Regression Trees

Predictor Space:



The goal is to minimize:

$$\sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \hat{y}_{R_j})^2,$$

Where $\hat{y}_{Rj}$ is the mean response for the training observations within the *j*th box.

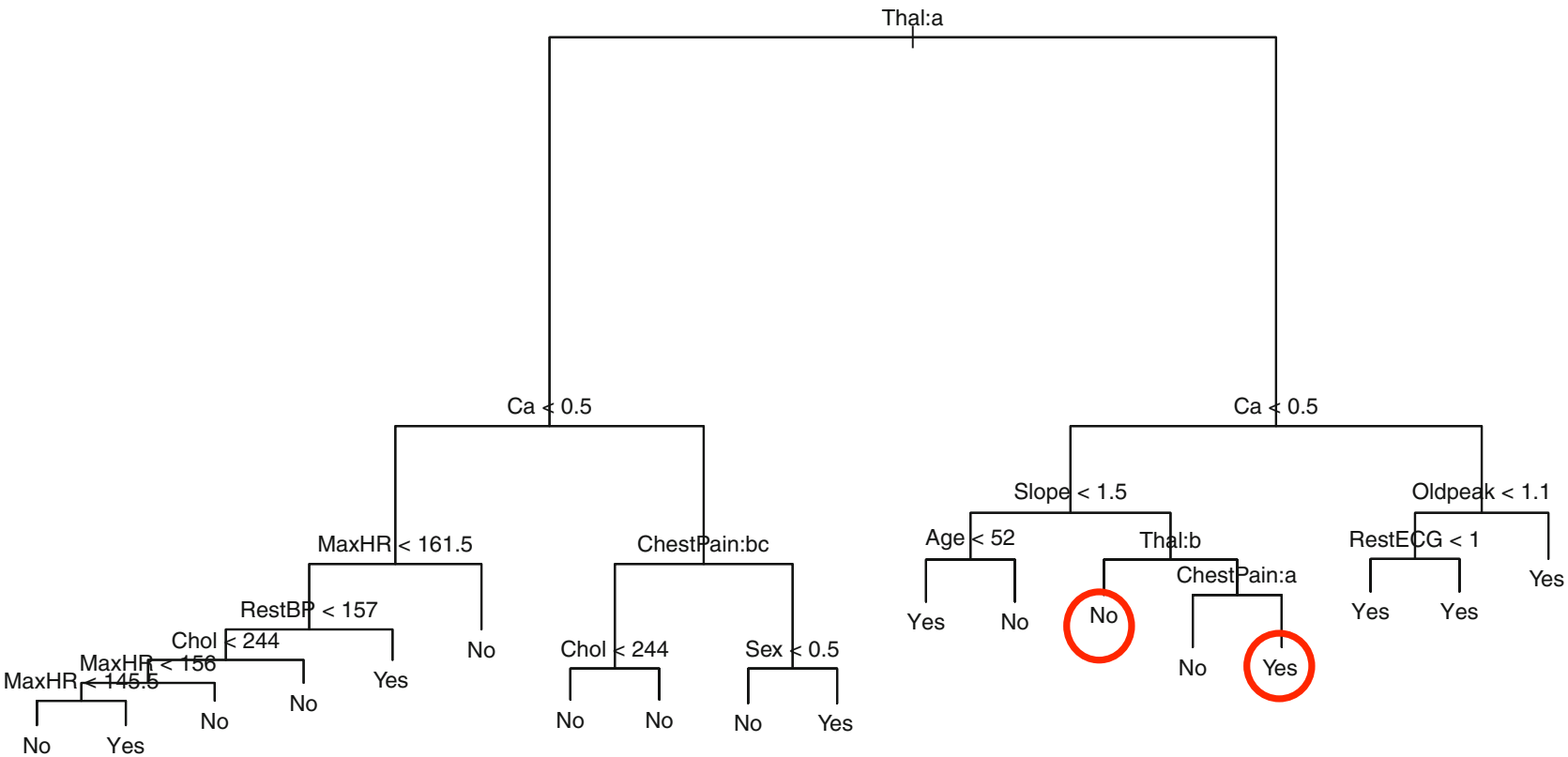
Details

- For computation reasons, the algorithm takes a top-down, greedy approach known as *Recursive Binary Splitting*

- The tree does not look ahead!!

Example: Classification tree

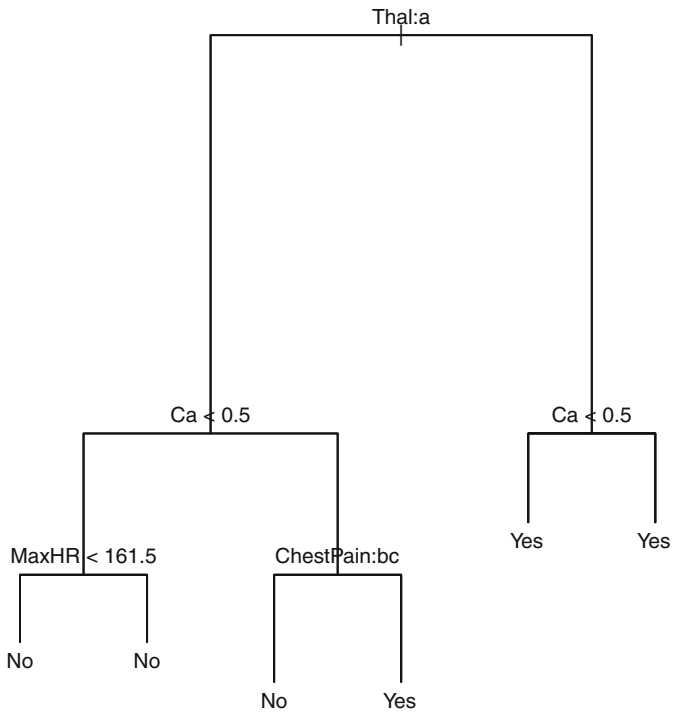
Does the patient have heart disease?



Classification trees have **Discrete** response variables

Example: classification tree

Pruned tree:



The first tree OVERFITS the training data

Bagging

- Regular trees suffer from high ***VARIANCE***
  - If we split our data into 2 parts and made a tree with each, we might get really different results
- We want to take the average predictions of many trees constructed from different samples of the population
  - We don't actually have different samples so we bootstrap them
- Bagging = ***Bootstrap Aggregation***
  - *Ensemble method!*

Issues with Bagging

- Bagging involves averaging many highly ***CORRELATED*** trees.
- Bagging does not lead to a substantial reduction in variance.
- Bagging is more popular with other learning algorithms.

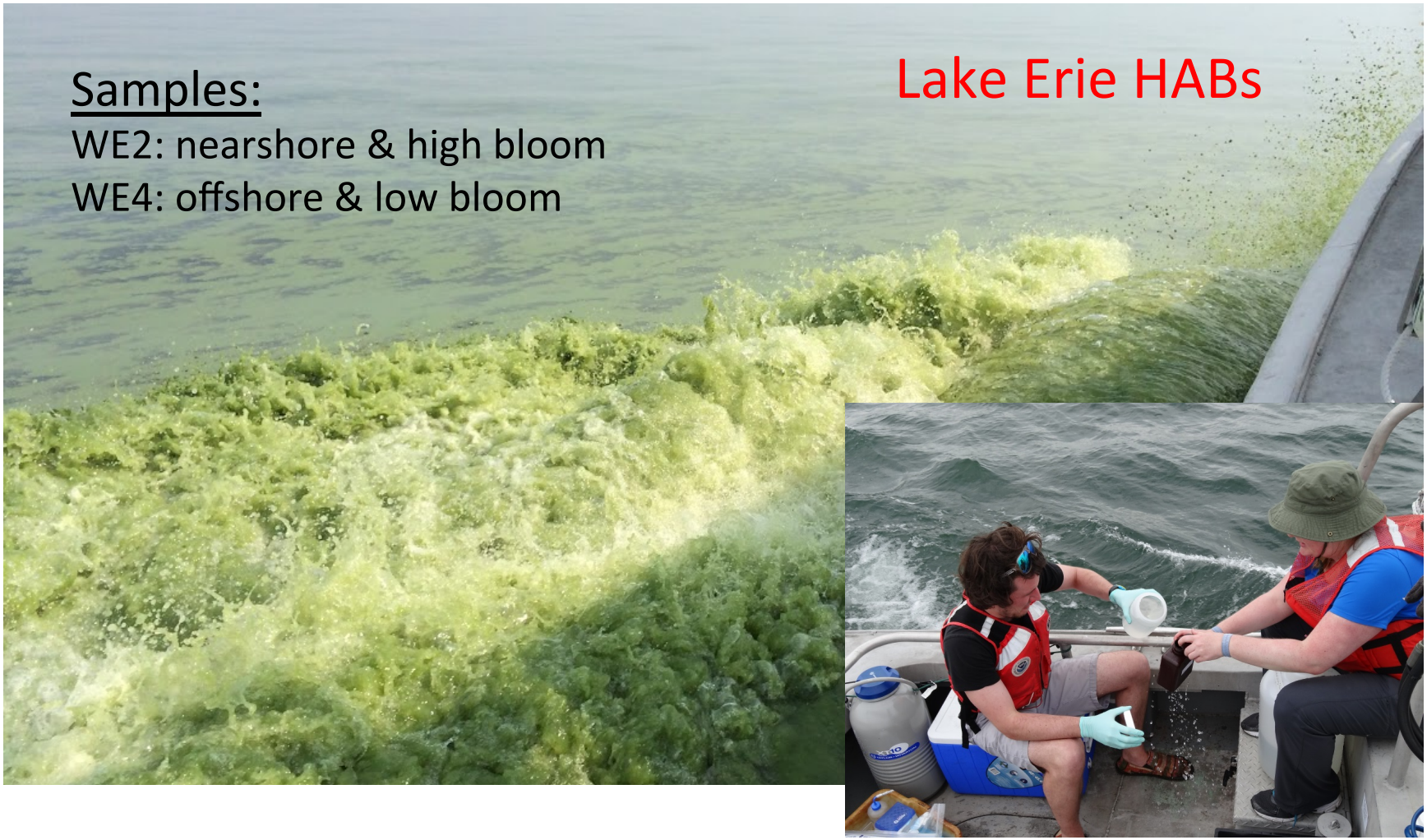
Random Forests

- Random forests address this problem by ***DECORRELATING*** the trees.

- As in Bagging we build many decision trees on Bootstrapped training samples, but when building these trees we only consider a subset m of all p predictors at each split.

- Usually $m = \sqrt{p}$

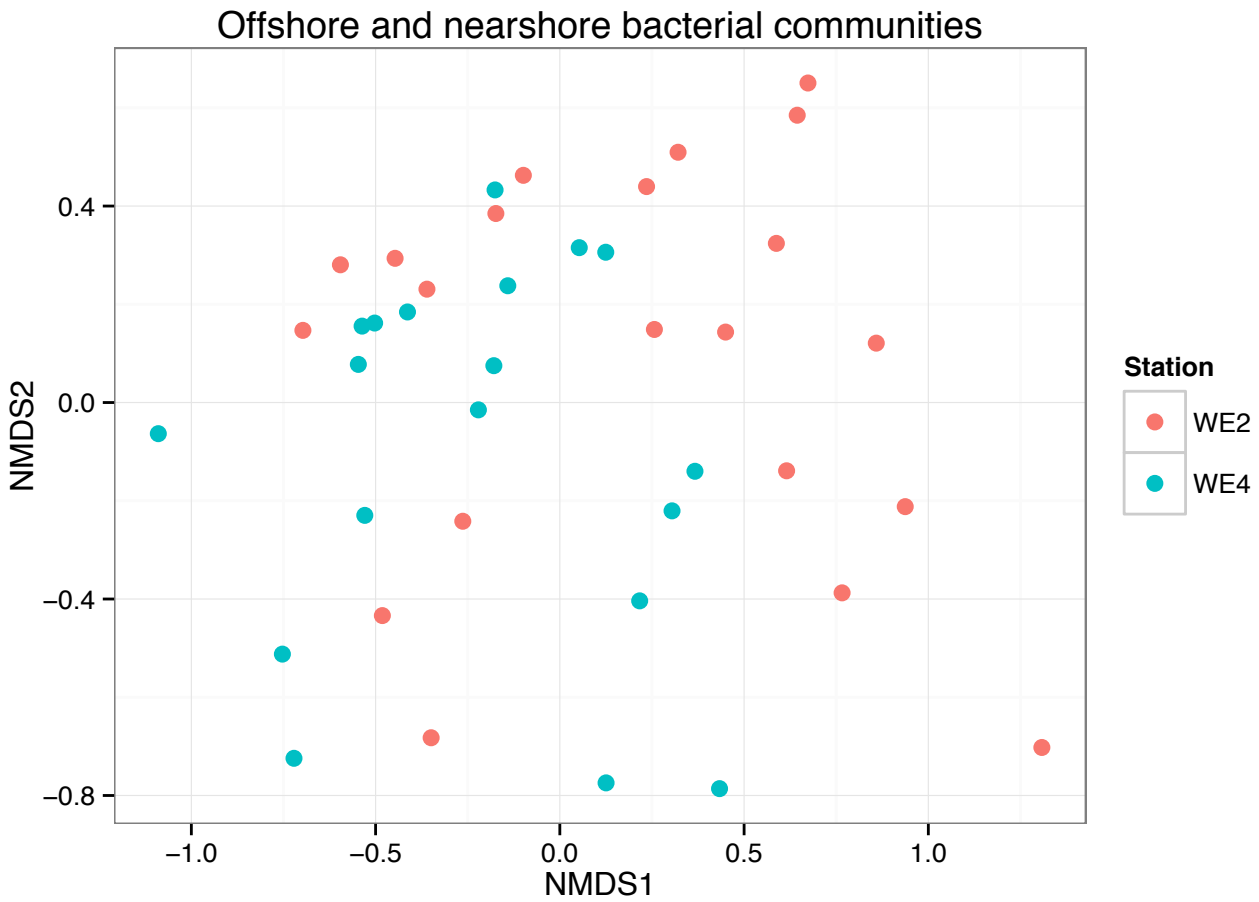
Microbial Ecology Applications



Example

Classification Random Forest
with
microbial community data

Goal: Use bacterial community composition to predict whether a sample comes from a nearshore (WE2) or offshore (WE4) site



Classification Random Forest

Goal: Use bacterial community composition to predict whether a sample comes from a near shore (WE2) or offshore (WE4) site.

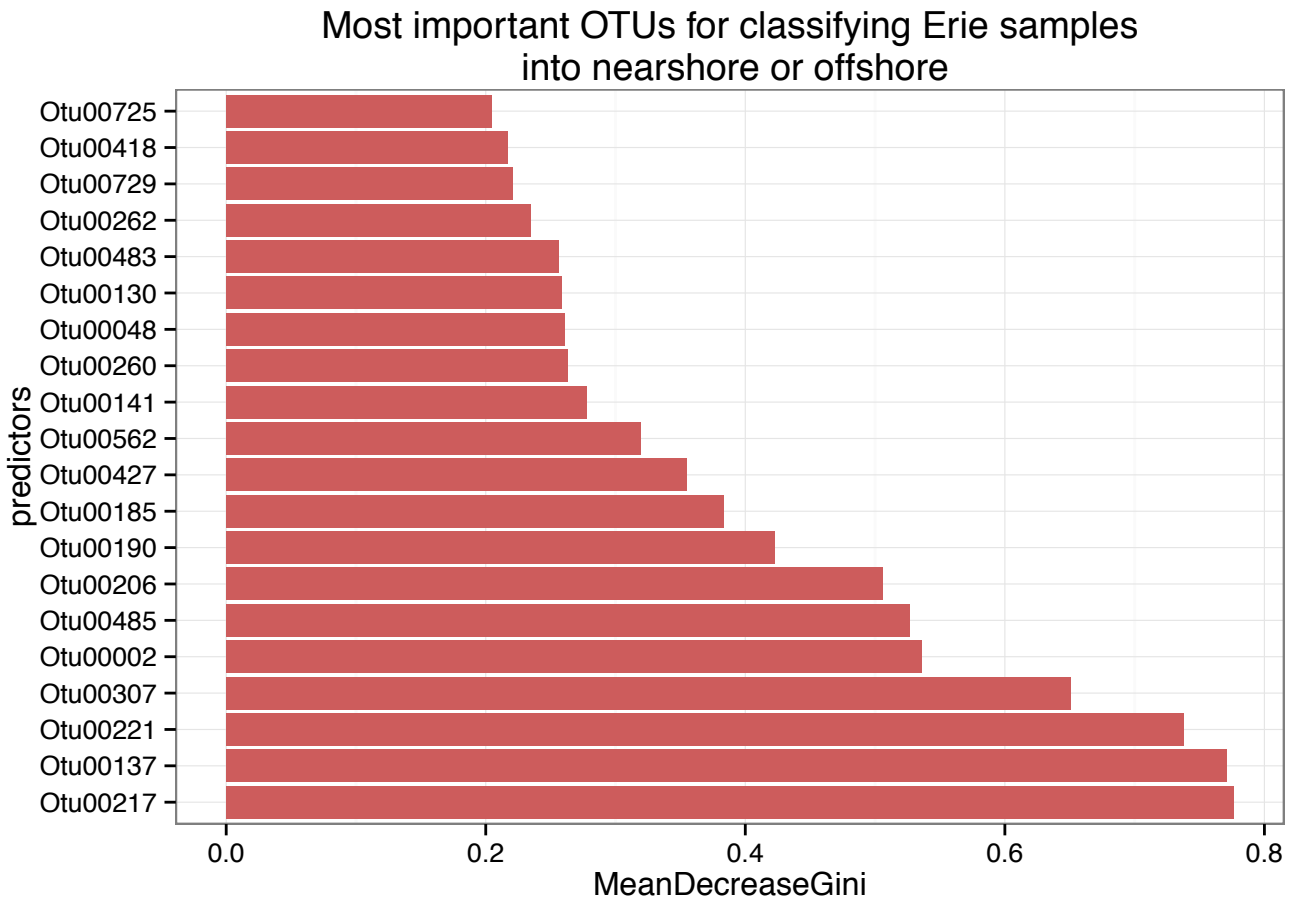
```
> set.seed(2)
> erie.classify <- randomForest(response~., data = rf.data, ntree = 100)
> print(erie.classify)

Call:
 randomForest(formula = response ~ ., data = rf.data, ntree = 100)
               Type of random forest: classification
                     Number of trees: 100
No. of variables tried at each split: 19

        OOB estimate of  error rate: 17.5%
Confusion matrix:
    WE2 WE4 class.error
WE2  18   3   0.1428571
WE4   4  15   0.2105263
>
```

"Out of Bag" error

Test set error rate: 17.5%


Most important OTUs for classifying Erie samples
into nearshore or offshore
predictors
Otu00725
Otu00418
Otu00729
Otu00262
Otu00483
Otu00130
Otu00048
Otu00260
Otu00141
Otu00562
Otu00427
Otu00185
Otu00190
Otu00206
Otu00485
Otu00002
Otu00307
Otu00221
Otu00137
Otu00217
0.0
0.2
0.4
0.6
0.8
MeanDecreaseGini

Recap

- Decision trees are a statistical learning algorithm
- Random forests are an improvement on the basic decision tree algorithm
- Random forests can help you classify or regress microbial communities and identify the OTUs most relevant to the characteristic you are modeling
- Random Forests are very easy to run in R

Resources

- James, G., Witten, D., Hastie, T., Tibshirani, R. A "An Introduction to Statistical Learning with Applications in R" (Springer, 2013)
  - Free online through Umich Library

- Stanford MOOC on Statistical Learning

https://class.stanford.edu/courses/HumanitiesandScience/StatLearning/Winter2015/about

Papers

- Knights, D., Costello, E. K., & Knight, R. (2011). “Supervised classification of human microbiota”. FEMS microbiology reviews, 35(2), 343-359.

- Beck, D., & Foster, J. A. (2014). “Machine learning techniques accurately classify microbial communities by bacterial vaginosis characteristics”. PloS one, 9(2), e87830.

- Xu, Z., Malmer, D., Langille, M., Way S., Knight, R. (2014) “Which is more important for classifying microbial communities: who’s there or what they can do”. The ISME Journal, 8, 2357-2359.

Want to chat?

Michelle Berry

Denef Lab

michberr@umich.edu