Lecture 4: Tree-based Methods

Hector Corrada Bravo and Rafael A. Irizarry

February, 2010

The next four paragraphs are from the book by Breiman et al.

At the University of California, San Diego Medical Center, when a heart attack patient is admitted, 19 variables are measured during the first 24 hours. They include BP, age, and 17 other binary covariates summarizing the medical symptoms considered as important indicators of the patient's condition.

The goal of a medical study can be to develop a method to identify high risk patients on the basis of the initial 24-hour data.

The next Figure shows a picture of a tree-structured classification rule that produced in the study. The letter F means not high-risk and the letter G means high risk.

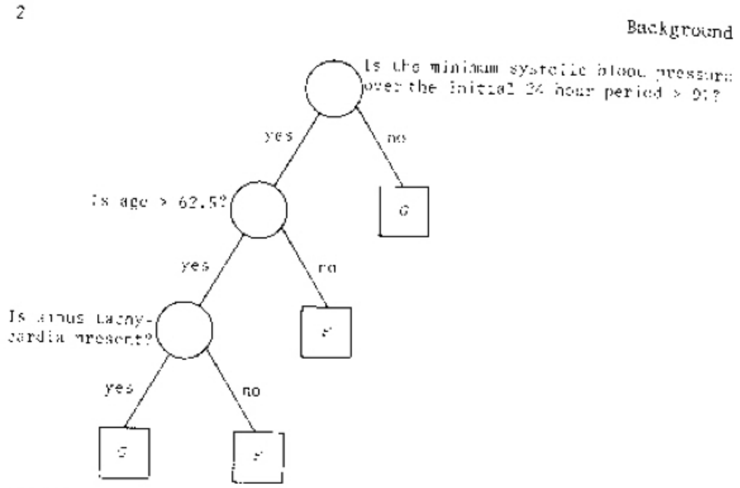

2
Background
Is the minimum systolic blood pressure over the initial 24 hour period > 91?
yes
no
G
Is age > 62.5?
yes
no
F
Is sinus tachycardia present?
yes
no
G
F

FIGURE 1.1

How can we use data to construct trees that give us useful answers. There is a large amount of work done in this type of problem. We will give an introductory description in this Section.

The material here is based on lectures by Ingo Ruczinski.

Classifiers as Partitions

Notice that in the previous example we predict a positive outcome if both blood pressure is high **and** age is higher than 62.5. This type of interaction is hard to describe in a regression model. So far, we have not discussed any methods that include interactions, mainly due to the curse of dimensionality. There are too many interactions to consider and too many ways to quantify their effect. Regression trees thrive on such interactions. What is a curse for parameteric approaches is a blessing for regression trees.

A good example is the following olive data:

- 572 olive oils were analyzed for their content of eight fatty acids (palmitic, palmitoleic, stearic, oleic, linoleic, arachidic, linolenic, and eicosenoic).
- There were 9 collection areas, 4 from Southern Italy (North and South Apulia, Calabria, Siciliy), two from Sardinia (Inland and Coastal) and 3 from Northern Italy (Umbria, East and West Liguria)
- The concentration of different fatty acids vary from up to 85% for oleic acid to as lows as 0.01% for eicosenic acid.
- See Forina M., Armanino C., Lanteri S., and Tiscornia E. (1983). *Classification of olive oils from their fatty acid composition.* In Martens H. and Russwurm Jr. H. eds., Food Research and Data Analysis, pp. 189–214. Applied Science Publishers, London

The data look like this:

Notice that we can separate the covariate space so that we get perfect prediction without a very complicated "model".

The tree representation of this picture is

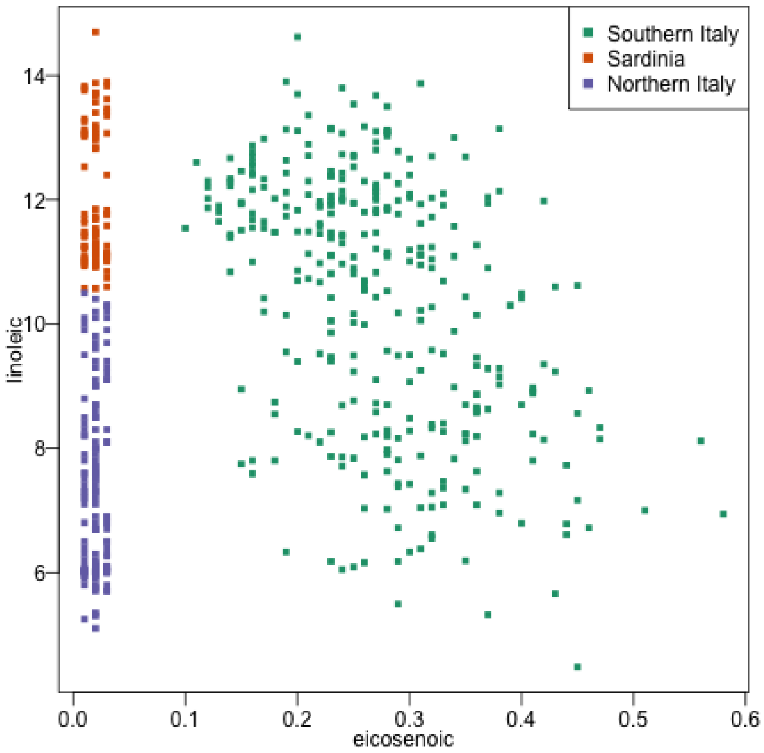
Partitions such as these can also handle data where linear methods work well. A good (and very famous) example is Fisher's Iris Data:

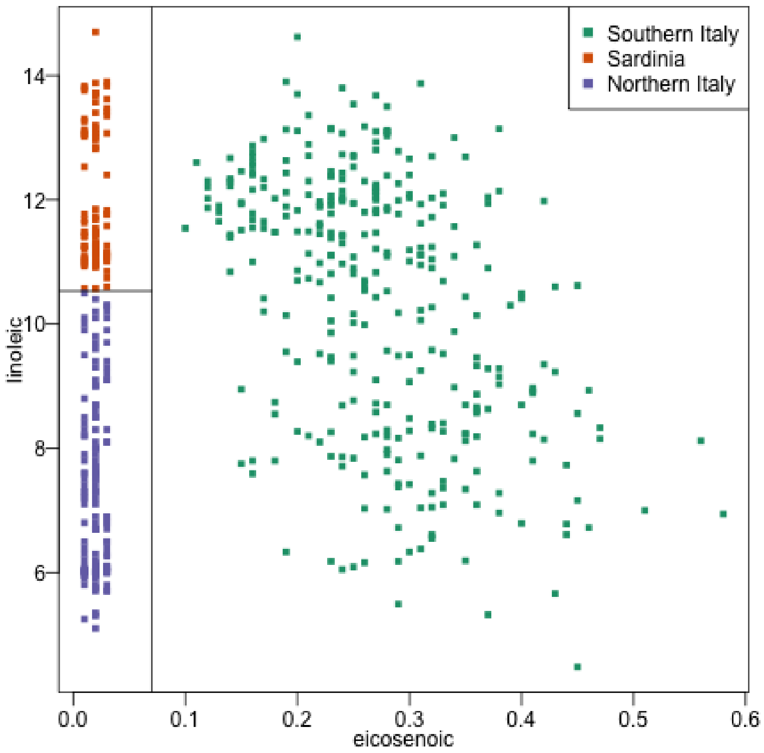
However, none of the methods we have described permit a division of the space without using many parameters.

Trees

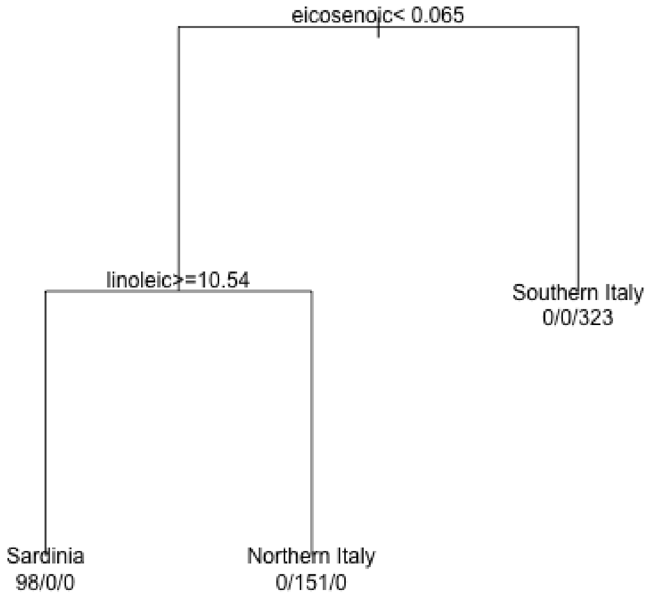
These data motivate the approach of partitioning the covariate space $\mathcal{X}$ into disjoint sets $A_1, \dots, A_j$ with $\hat{G} = j$ for all $\mathbf{x} \in A_j$. There are too many ways of doing so we try to make the approach *parsimonious*. Notice that linear regression/classification restrict partition to certain planes.

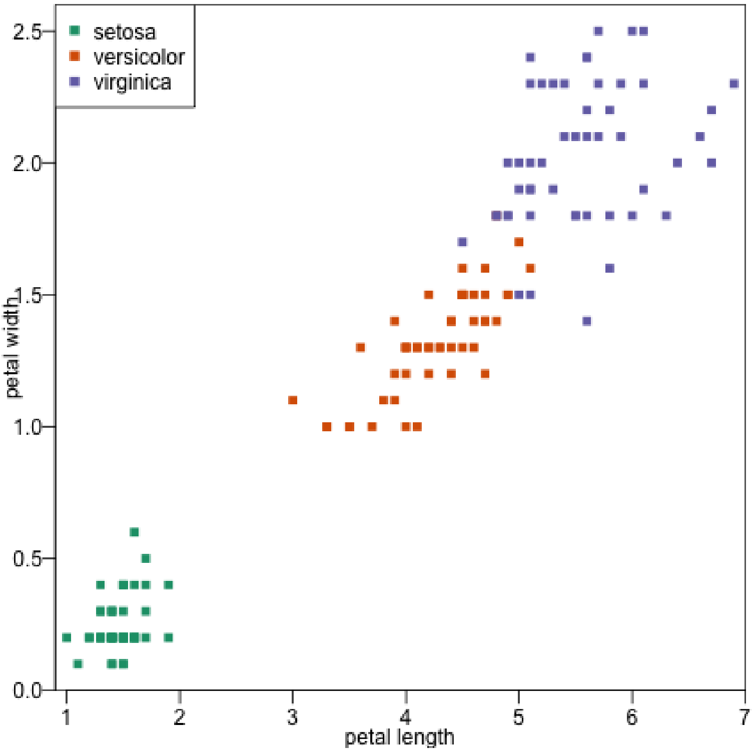
Trees are a completely different way of partitioning. All we require is that the partition can be achieved by successive binary partitions based on the different


Southern Italy
Sardinia
Northern Italy
14
12
10
8
6
linoleic
0.0
0.1
0.2
0.3
0.4
0.5
0.6
eicosenoic

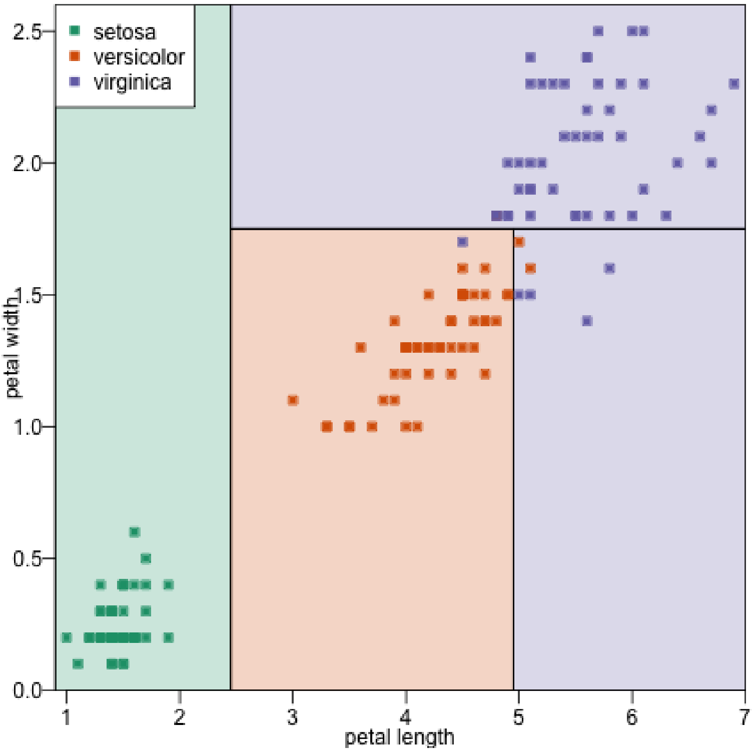


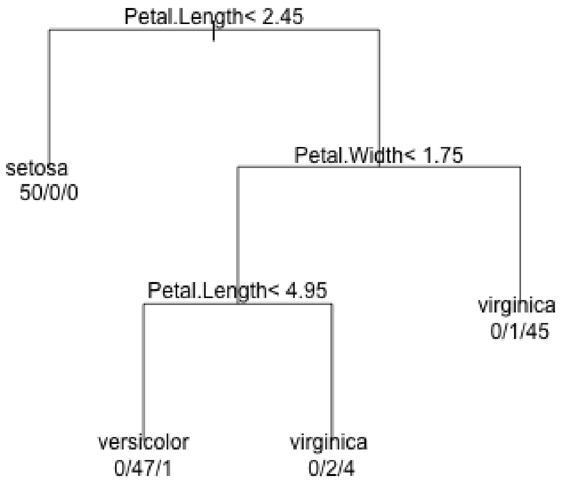
Southern Italy
Sardinia
Northern Italy
14
12
10
8
6
linoleic
0.0
0.1
0.2
0.3
0.4
0.5
0.6
eicosenoic


eicosenoic< 0.065
linoleic>=10.54
Southern Italy
0/0/323
Sardinia
98/0/0
Northern Italy
0/151/0



setosa
versicolor
virginica
2.5
2.0
1.5
1.0
0.5
0.0
petal width
1
2
3
4
5
6
7
petal length

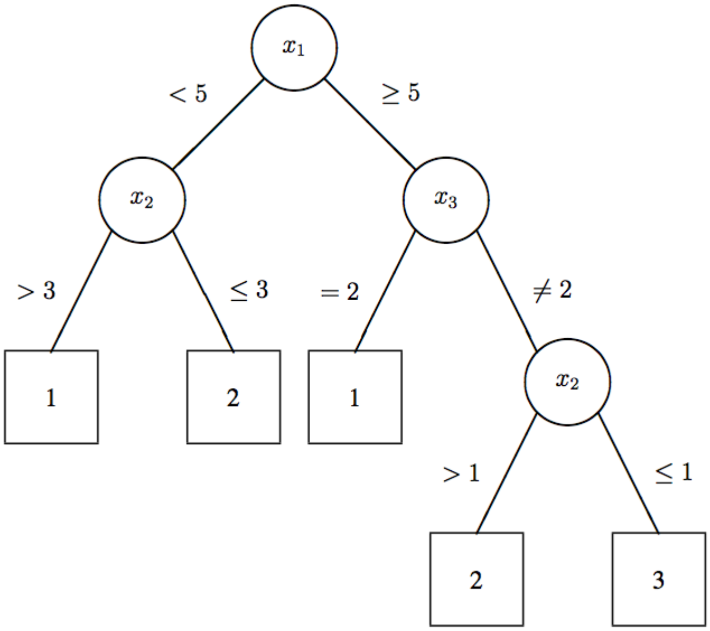

setosa
versicolor
virginica
2.5
2.0
1.5
1.0
0.5
0.0
petal width
1
2
3
4
5
6
7
petal length


Petal.Length< 2.45
setosa
50/0/0
Petal.Width< 1.75
Petal.Length< 4.95
virginica
0/1/45
versicolor
0/47/1
virginica
0/2/4

predictors. Once we have a partition, we base our prediction on the average of the Ys in each partition. We can use this for both classification and regression.

Example of a classification tree

Suppose that we have a scalar outcome, Y, and a p-vector of explanatory variables X. Assume $Y \in \mathcal{K} = \{1, 2, \ldots, k\}$.



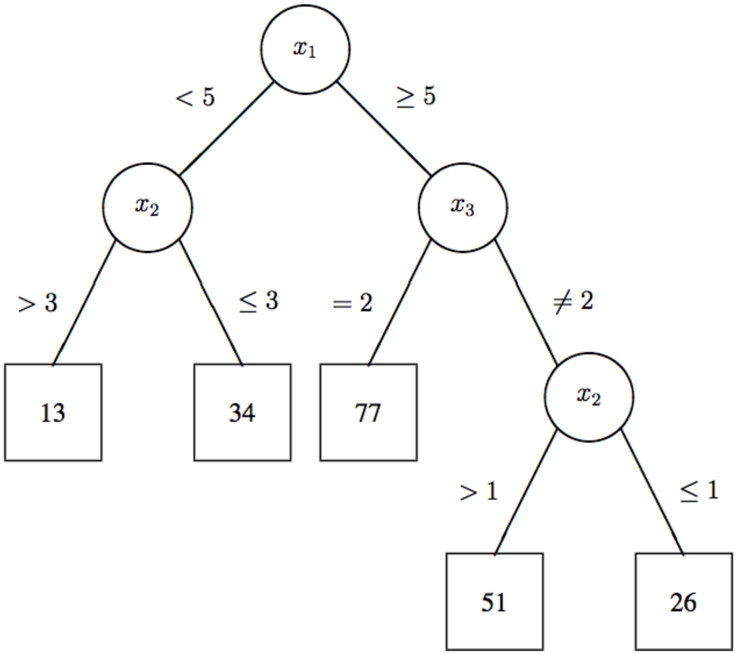
The subsets created by the splits are called *nodes*. The subsets which are not split are called *terminal nodes*.

Each terminal node gets assigned to one of the classes. So if we had three classes, we could get $A_1 = \mathcal{X}_1 \cup \mathcal{X}_9, A_2 = \mathcal{X}_6$ and $A_3 = \mathcal{X}_7 \cup \mathcal{X}_8$. If we are using the data, we assign the class most frequently found in that subset of $\mathcal{X}$. We call these *classification trees*.

A classification tree partitions the X-space and provides a predicted value, perhaps $\arg\max_s Pr(Y = s|X \in A_k)$ in each region.

Example of a regression tree

Again, suppose we have a scalar outcome Y, and a p-vector of explanatory variables X. Now assume $Y \in \mathbb{R}$.



CART versus Linear Models

See Figure

Searching for good trees

In general, the idea is the following:

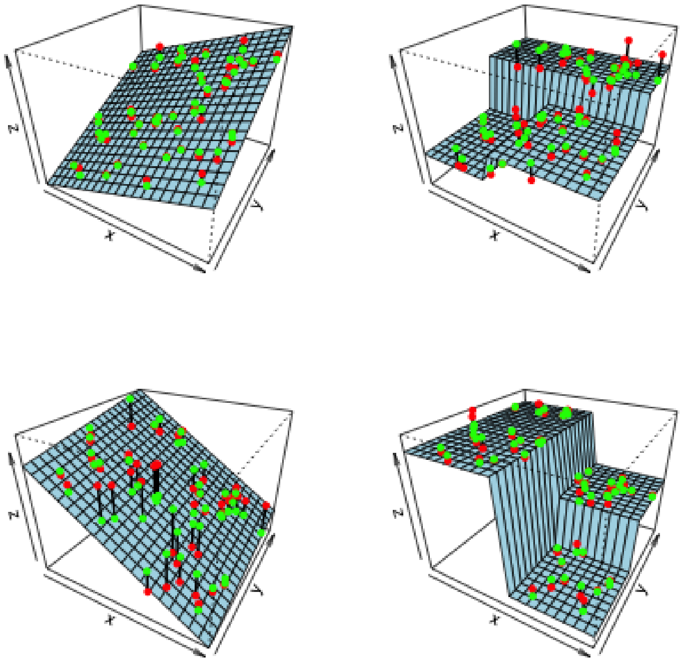
1. **Grow** an overly large tree usign forward selection. At each step, find the *best* split. Grow until all terminal nodes either

   (a) have $< m$ (perhaps $m = 1$) data points

   (b) are "pure" (all points in a node have [almost] the same outcome).

2. **Prune** the tree back, creating a nested sequence of trees, decreasing in *complexity*

A problem in tree construction is how to use the training data to determine the binary splits of $\mathcal{X}$ into smaller and smaller pieces. The fundamental idea is to select each split of a subset so that the data in each of the descendent subsets are "purer" that the data in the parent subset.


z
x
y
z
x
y
z
x
y
z
x
y

The predictor space

Suppose we have p explanatory variables $X_1, \dots, X_p$ and N observations.

Each of the X_i can be

a) a numeric variable: $\rightarrow n-1$ possible splits

b) an ordered factor (categorical variable): $\rightarrow k-1$ possible splits

c) an unordered factor: $\rightarrow 2^{k-1}-1$ possible splits.

We pick the split that results in the greatest decrease in *impurity*. We will soon provide various definitions of impurity.

Deviance as a measure of impurity

A simple approach is to assume a multinomial model and then use deviance as a definition of impurity.

Assume $Y \in \mathcal{G} = \{1, 2, \dots, k\}$.

- At each node i of a classification tree we have a probability distribution p_{ik} over the k classes.
- We observe a random sample n_{ik} from the multinomial distribution specified by the probabilities p_{ik}.
- Given X, the conditional likelihood is then proportional to $\prod_{(\text{leaves } i)} \prod_{(\text{classes } k)} p_{ik}^{n_{ik}}$.
- Define a deviance $D = \sum D_i$, where $D_i = -2 \sum_k n_{ik} \log(p_{ik})$.
- Estimate p_{ik} by $\hat{p}_{ik} = \frac{n_{ik}}{n_i}$.

For the olive tree we get the following values:

Root	$n_{11} = 323$	$n_{12} = 98$	$n_{13} = 151$	$n_1 = 572$	D=1117.18
	$\hat{p}_{11} = 0.56$	$\hat{p}_{12}$=0.17	$\hat{p}_{13} = 0.24$		
Split 1	$n_{11} = 323$	$n_{12} = 0$	$n_{13} = 0$	$n_1 = 323$	$D = 333.82$
	$n_{21} = 0$	$n_{22} = 98$	$n_{23} = 151$	$n_2 = 249$	
	$\hat{p}_{11} = 1$	$\hat{p}_{12} = 0$	$\hat{p}_{13} = 0$		
	$\hat{p}_{21} = 0$	$\hat{p}_{22} = 0.39$	$\hat{p}_{23} = 0.41$		
Split 2	$n_{11} = 323$	$n_{12} = 0$	$n_{13} = 0$	$n_1 = 323$	$D = 0$
	$n_{21} = 0$	$n_{22} = 0$	$n_{23} = 151$	$n_2 = 151$	
	$n_{31} = 0$	$n_{32} = 98$	$n_{33} = 0$	$n_3 = 98$	

Other measures of impurity

Other commonly used measures of impurity at a node i of a classification tree are

- missclasification rate: $\frac{1}{n_i} \sum_{j \in A_i} I(y_j \neq k_i) = 1 - \hat{p}_{ik_i}$.
- the entropy: $\sum p_{ik} \log(p_{ik})$
- the GINI index: $\sum_{j \neq k} p_{ij} p_{ik} = 1 - \sum_k p_{ik}^2$

where k_i is the most frequent class in node i.

For regression trees we use the residual sum of squares:

$$D = \sum_{\text{cases } j} (y_j - \mu_{[j]})^2$$

where $\mu_{[j]}$ is the mean values in the node that case j belongs to.

Recursive partitioning

INITIALIZE All cases in the root node
REPEAT Find optimal allowed split
Partition leaf according to split
STOP Stop when pre-defined criterion is met

Model Selection

- Grow a big tree T
- Consider snipping off terminal subtrees (resulting in so-called rooted subtrees)
- Let R_i be a measure of impurity at leaf i in a tree. Define $R = \sum_i R_i$
- Define size as the number leaves in a tree
- Let $R_\alpha = R + \alpha \times \text{size}$

The set of rooted subtrees of T that minimize R_α is nested.

General Points

What's nice:

- Decision trees are very "natural" constructs, in particular when the explanatory variables are catgorical (and even better when they are binary)
- Trees are easy to explain to non-statisticians
- The models are invariant under transformations in the predictor space
- Multi-factor responses are easily dealt with
- The treatment of missing values is more satisfactory than for most other models
- The models go after interactions immediately, rather than as an afterthought
- Tree growth is much more efficient than described here
- There are extensions for survival and longitudinal data, and there is an extension called treed models. There is even a Bayesian version of CART

What's not so nice

- Tree space is huge, so we may need lots of data
- We might not be able to find the *best* model at all
- It can be hard to assess uncertainty in inference about trees
- Results can be quite variable (tree selection is not very
- stable)
- Actual additivity becomes a mess in a binary tree
- Simple trees usually don't have a lot of predictive power
- There is a selection bias for the splits

CART references

- L. Breiman. *Statistical Modeling: The Two Cultures.* Statistical Science, 16(3), pp 199–215, 2001.
- L. Breiman, JH Friedman, RA Olshen, and CJ Stone. *Classification and Regression Trees.* Wadsworth Inc., 1984.

- TM Therneau and EJ Atkinson. *An Introduction to Recursive Partitioning Using the RPART Routines.* Technical Report Series, No. 61, Department of Health Science Research, Mayo Clinic, Rochester, Minnesota, 2000.

- WM Venables, and BD Ripley. *Modern Applied Statistics with S* Springer, NY, 4th edition, 2002.