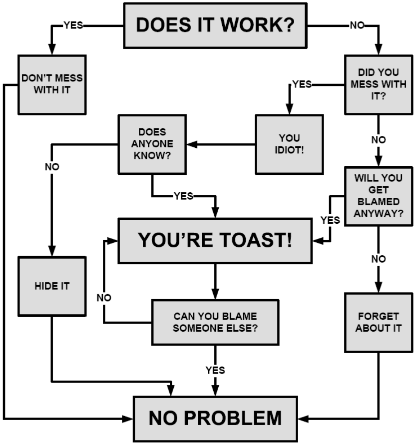
Data, Environment and Society: Lecture 16: Regression trees

Instructor: Duncan Callaway
GSI: Salma Elmallah

October 24, 2019

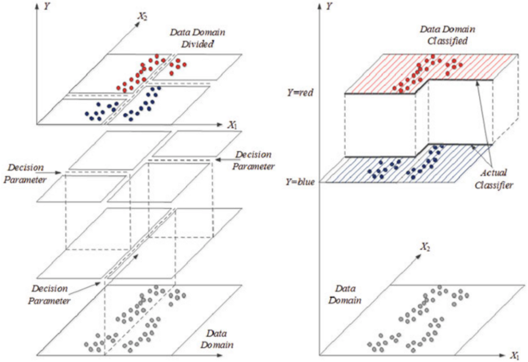
Problem Solving Flowchart



https://thenexttobestblogever.wordpress.com/2009/11/07/problem-solving-flowchart-2/

Objectives

- Introduction to regression trees
  - Terminology
  - How they are built
  - How to choose with cross validation
- Next week, we'll discuss classification trees
  - Same as regression, just different loss functions



(medium.com)

Terminology we'll cover...

- Terminal node
- Internal node
- Branches
- Leaves
- Binary splits
- Recursive binary splitting $\leftrightarrow$ Top-down greedy
- Cost complexity pruning

Basic idea for regression trees

All we are doing is "splitting" the observations into regions in the predictor space, and averaging the response variable within each region.

Basic idea for regression trees

All we are doing is "splitting" the observations into regions in the predictor space, and averaging the response variable within each region.

Doing predictions with the model just involves

- locating a set of **predictors** in a region,
- then setting the response variable equal to the average from the training **response** variable in that region.

Basic idea for regression trees

All we are doing is "splitting" the observations into regions in the predictor space, and averaging the response variable within each region.

Doing predictions with the model just involves

- locating a set of **predictors** in a region,
- then setting the response variable equal to the average from the training **response** variable in that region.

Big decision in regression trees: *What are the regions we should use?*

Example, from the textbook

“Hitters” data from ISLR.

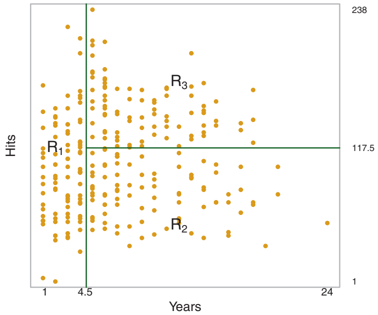
263 major league players stats.

Here, this tree is “spliting” on two variables – years in league and number of hits

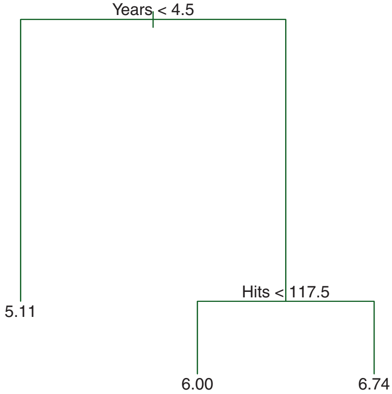
The numbers at the ends are the average (log-transformed) average salaries for players

Example, from the textbook, ctd

$$R_1 = \{X | \text{years} < 4.5\}$$
$$R_2 = \{X | \text{years} \geq 4.5, \text{hits} < 117.5\}$$
$$R_3 = \{X | \text{years} \geq 4.5, \text{hits} \geq 117.5\}$$



Terminology



Each region R_i is a *terminal node*

Each numeric value at which a split happens is an *internal node*

Segments connecting nodes (terminal or internal) are...

Terminology

Each region R_i is a *terminal node*

Each numeric value at which a split happens is an *internal node*

Segments connecting nodes (terminal or internal) are...*branches*

Terminology

Each region R_i is a *terminal node*

Each numeric value at which a split happens is an *internal node*

Segments connecting nodes (terminal or internal) are...*branches*

The numbers at the end of the branches are also sometimes called...

Terminology

Each region R_i is a *terminal node*

Each numeric value at which a split happens is an *internal node*

Segments connecting nodes (terminal or internal) are...*branches*

The numbers at the end of the branches are also sometimes called...*leaves*

Terminology so far...

- Terminal node
- Internal node
- Branches
- Leaves

Regression trees – basic approach

1. Divide the **predictor** space into non-overlapping regions
   - This distinguishes the method from KNN regression
2. Within each region, the prediction is just the average of the **response variable** from training data.
   - This is similar to KNN regression

Regression trees – basic approach

1. Divide the **predictor** space into non-overlapping regions
   - This distinguishes the method from KNN regression
2. Within each region, the prediction is just the average of the **response variable** from training data.
   - This is similar to KNN regression

Two Basic Questions:

1. Where should I put the internal nodes?
2. How many regions should there be?

The answers are, as it turns out, really simple.

Where to put the internal nodes?

First, for simplicity, the nodes are structured to make rectangles in the a 2-D predictor space (or hyper-rectangles in higher dimensions).

How do I split regions?

Let

- j index predictor variables
- s denote the location of the split within the region
    - (With n observations we have to consider at most $n-1$ split points; the numeric value of the split is the mid-way point between to adjacent observations.)

Then all splits can be described as:

$$R_1(j,s) = \{X|X_j < s\} \text{ and } R_2(j,s) = \{X|X_j \geq s\}$$

But where should the splits be?

Then we partition any region by choosing j and s as follows:

$$\{j, s\} = \arg \min_{j \in J, s \in X_j} \sum_{i: x_i \in R_1(j,s)} (y_i - \hat{y}_{R_1})^2 + \sum_{i: x_i \in R_2(j,s)} (y_i - \hat{y}_{R_2})^2$$

where $\hat{y}_{R_k}$ is the mean of all response variables in region k.

It would be tedious to identify j and s by hand, but it's actually very quick computationally. (Remember, there are only $n - 1$ possible splits for each predictor.)

Ok, we've split one predictor in two. Now what?

Next choose the single best split from among *all* possible splits of the two new regions. **Now we'll have three regions.**

In general, on the n^{th} step, choose the single best possible split from among the n regions, resulting in $n+1$ regions to take to the next step.

Repeating the splits

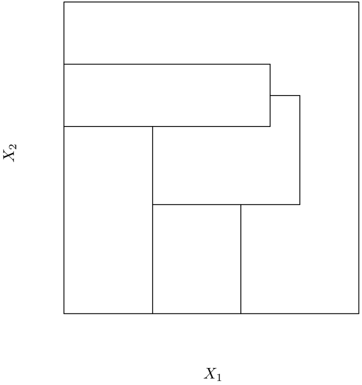
On each step, we're choosing the single best possible split from among the n regions, resulting in $n + 1$ regions to take to the next step.

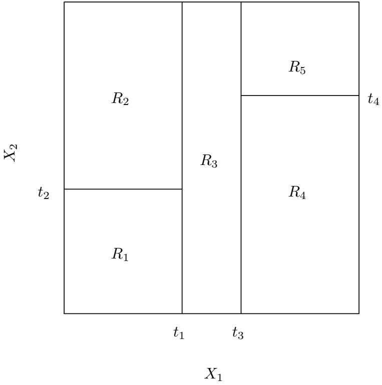
Repeat this process until you reach a stopping criterion – typically a maximum number of observations in each region. (For example all regions have no more than 5 observations.)

Call the resulting tree T_0.

We call this approach "greedy" because when we do the first partition we're not thinking ahead to future partitions to evaluate it.

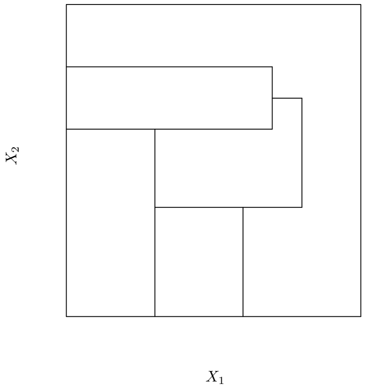
One of these doesn't belong...

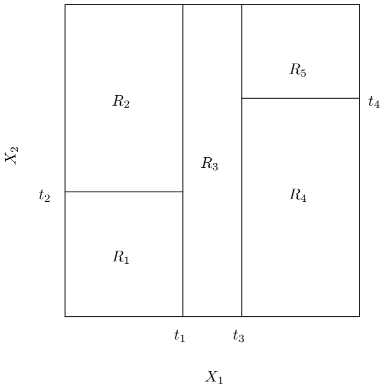




Q: Which picture results from successively splitting the regions into values greater or less than predictor values?

One of these doesn't belong...

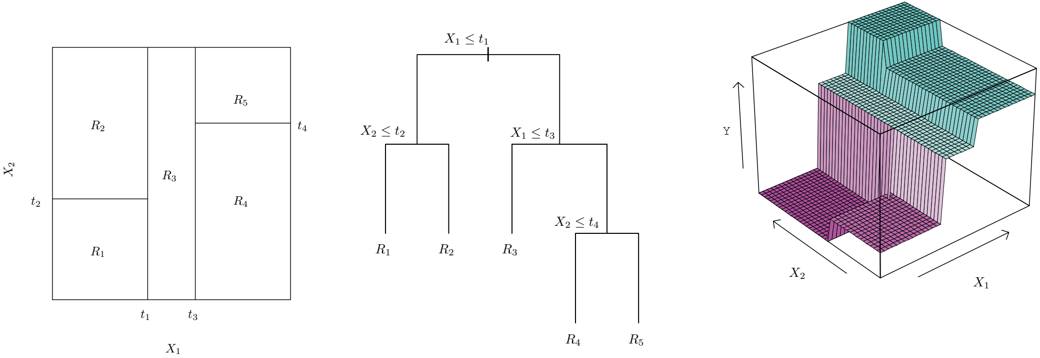




Q: Which picture results from successively splitting the regions into values greater or less than predictor values?

A: The right one. The left one is not possible with simple splitting.

A five region example... with two dimensional predictor space



What do we call it?

The process of splitting regions over and over is called...

"recursive binary splitting"

You can also call it a **"top-down greedy'**' approach.

Because it's "greedy" we can't be sure that the splits we're getting are the best possible splits.

Why binary?

In other words, why not multiway splits?

Why binary?

In other words, why not multiway splits?

In general multiway splits fragment the data too quickly, leaving insufficient data at the next level down

Since we do the binary splitting recursively, we get the same flexibility as a multiway split, since a region can be split a second time later.

Terminology so far...

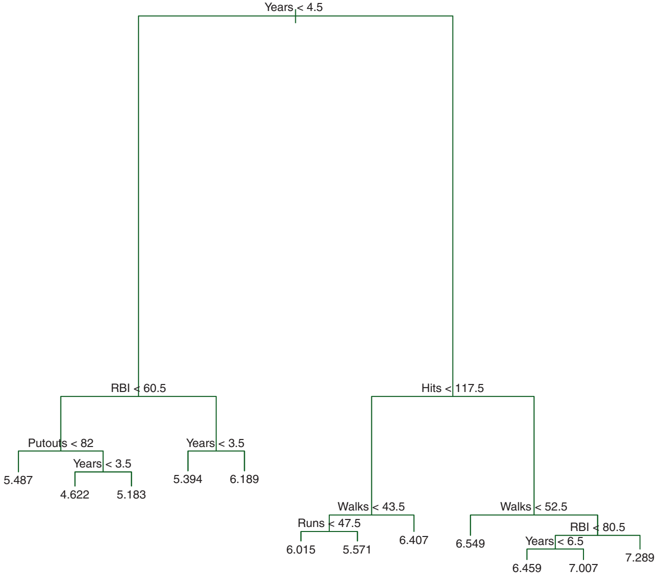
- Terminal node
- Internal node
- Branches
- Leaves

Terminology so far...

- Terminal node
- Internal node
- Branches
- Leaves
- Binary splits
- Recursive binary splitting $\leftrightarrow$ Top-down greedy

Example T_0

Remember, T_0 is the biggest tree we build. We get there by recursively splitting until we meet a threshold (often a maximum number of observations per terminal node).



See lecture 18 for finish up on decision trees.