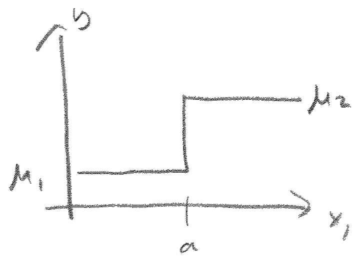
CART

Classification & Regression Trees

Idea: partition x-space into rectangular regions where $E(y)$ takes on a constant value in each region.

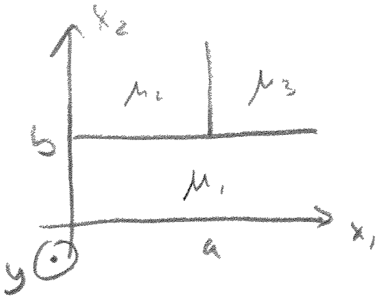
Examples.

One x-variable



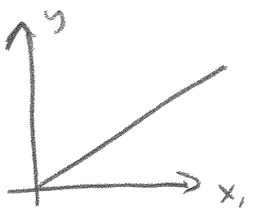
$$E(y) = \mu_1 \quad \text{if} \quad x_1 \le a$$
$$= \mu_2 \quad \text{if} \quad x_1 > a$$

2 variables



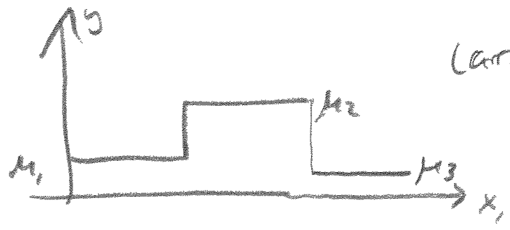
$$E(y_i) = \mu_1 \quad \text{if} \quad x_2 \le b$$
$$= \mu_2 \quad \text{if} \quad x_2 > b,\ x_1 \le a$$
$$= \mu_3 \quad \text{if} \quad x_2 > b,\ x_1 > a$$

Note



linear model
$E(y) = x\beta$

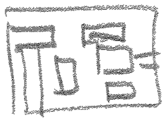
vs



Cart model

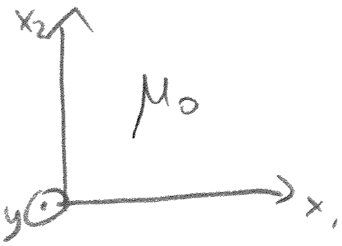
CART - allows for complex nonlinear dependencies.

How many rectangels / regions to use = new model selection problem.

Too many parameters to search over if we allow for any number & shape of regions 

So we use **Forward Search**

(0)

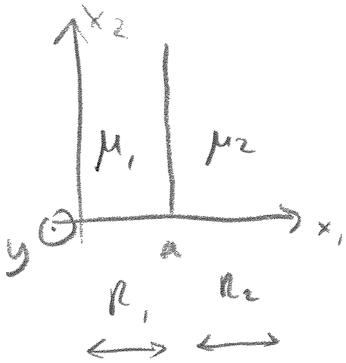


w/o any split on x-s

μ_0 best estimated by mean of $y = \bar{y}$

$$MSE_0 = \frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y})^2$$

(1) Consider splitting on x_1 at a



Estimate $\hat{\mu}_1$ = mean of y in region R_1

$$= \frac{\sum_{i=1}^{n} y_i \, 1\{x_{i1} \le a\}}{\sum_{i=1}^{n} 1\{x_{i1} \le a\}}$$

$$\hat{\mu}_2 = \frac{\sum_{i=1}^{n} y_i \, 1\{x_{i1} > a\}}{\sum_{i=1}^{n} 1\{x_{i1} > a\}}$$

$$\left(1\{x_i \le a\} = \begin{cases} 1 & \text{if } x_i \le a \\ 0 & \text{o/w} \end{cases} \right)$$

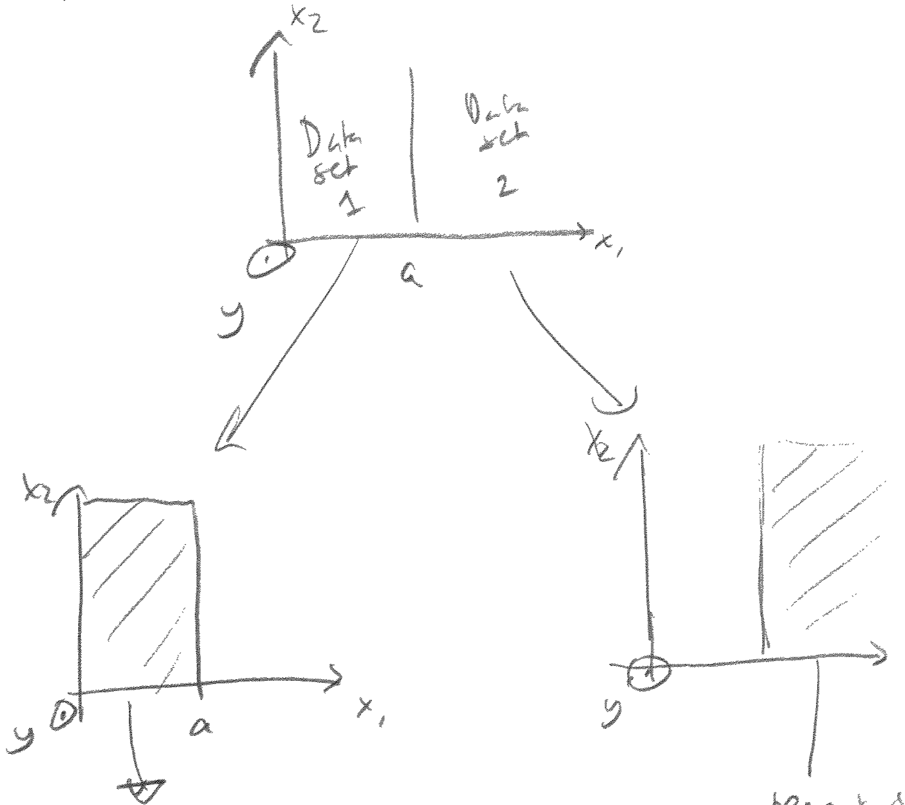
indicator

$$MSE_1(a) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{\mu}_1 1\{x_{i1} \le a\} - \hat{\mu}_2 1\{x_{i1} > a\}\right)^2$$

* Search over a to minimize MSE

* Search over x's to find best variable to split on

2. So let's say MSE was most reduced when we split on x_1 → form two data sets



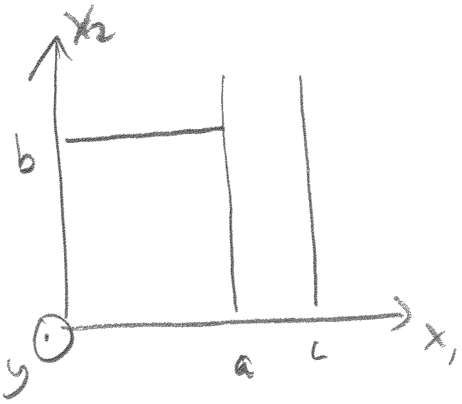
repeat step 1 for this data set

For this dataset

- repeat step 1

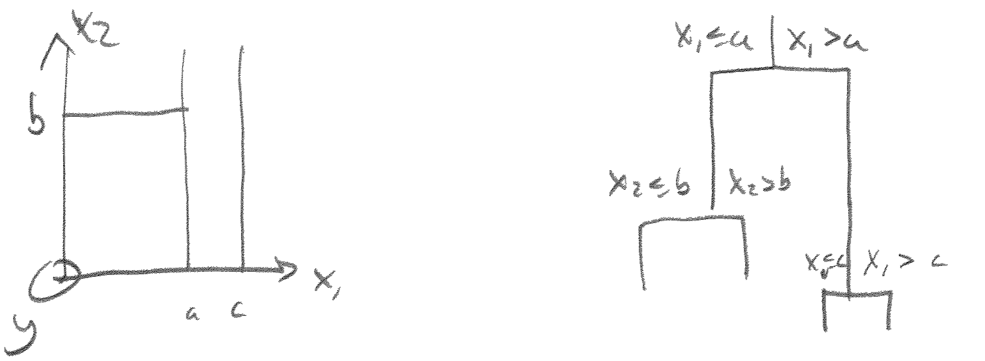
→ which variable to split on
→ and where

⇒



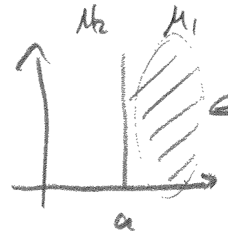
Here, for data set 1 the best split was on X_2 at b, and for data set 2 the best split was X_1 at c

We can display this model as a tree (4)



length of branch
= reduction in MSE
by doing this split

In this example



for this data

$$\text{MSE before} = \sum_{i=1}^{n} \left(y_i 1\{x_{i1} > a\} - \hat{\mu}_0\right)^2 \qquad (\hat{\mu}_0 = \bar{y})$$

$$\text{MSE after} = \sum_{i=1}^{n} \left(y_i 1\{x_i > a\} - \hat{\mu}_1\right)^2$$

length of branch $\approx$ MSE$_{\text{before}}$ − MSE$_{\text{after}}$

If MSE much reduced – split improved the fit a lot

How many splits to do?

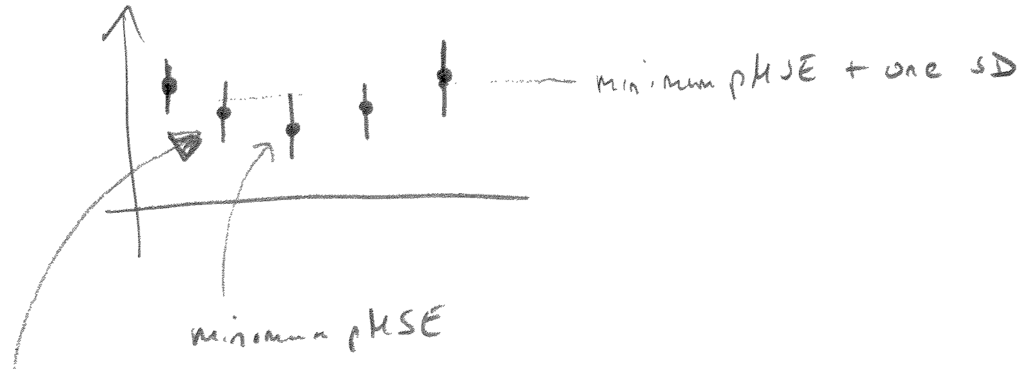
- Alt 1: Use F-test to decide if reduction of MSE after split large enough

- Alt 2: **Cross-Validation** → implemented in R.

Do CV: On each model (where a number of splits have been "snipped") compute the pMSE on each CV test set.

Compute the mean pMSE and its standard deviation

For each size tree (best tree of each size) plot mean pMSE ± SD



Pick smallest model with pMSE < minpMSE + SD

This model is as good as the one with minimum pMSE (overlapping pMSE intervals) so use the most simple one.

See Demo

Final remarks

- CART is "unstable" → on different random parts of data → trees we get from fitting can be very different!

→ Do like lab 3 → on random sets of data, fit trees & do CV selection → repeat many times

Which variables are always picked?

each tree → one prediction model } Use an average prediction as final prediction

(This is called BAGGING)