(1) Beginning with all training data....
(2) For every possible split at the current node divide the data into $X_L, \vec{y}_L$ & $X_R, \vec{y}_R$ and calculate

$$SSE_L = \sum (y_L - \bar{y}_L)^2 \quad , \quad SSE_R = \sum (y_R - \bar{y}_R)^2$$

(over # of data pts under the split)

(3) Find the split with the lowest Total SSE (sorry, lost the make records)

$$SSE_{TOT} = SSE_L + SSE_R$$

(4) Create the split, split data into two nodes
(5) Repeat steps 2-4 until "STOP"

STOP: node has $\leq N_0$ data pts inside.
Default: $N_0 = 5$

(6) For all leaf nodes, assign $\hat{y} = \bar{y}_0$ where $\bar{y}_0$ is the avg of the y's in that node.

There are many, many variations of the above.

If $N_0 = 1$ ... tree is grown to fit a separate parameter for each data pt. $R^2 = ?$ 100%! Then tree is pruned back to not overfit. Kind of like backwards stepwise regression.

How is N_0 picked? Model selection procedure via cv

DEMO

Classification!

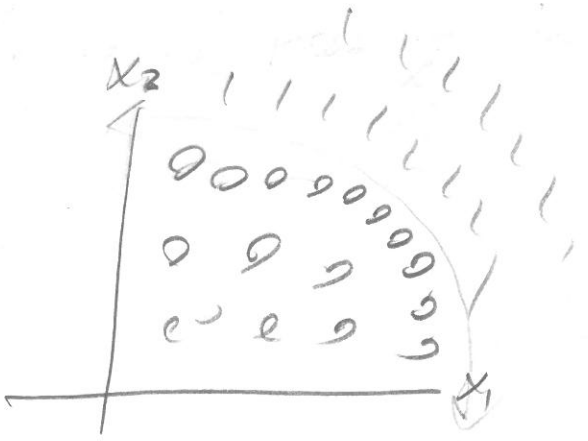
Remember $y = \{0,1\}$? Binary classification

or $y \in \{1,2,\dots,K\}$ general "multi" K groups / labels.

What methods do we have? For binary classification...

- perceptron
- SVM with hinge loss

} linear models → could we use kernels, polynomials, general functions? Yes

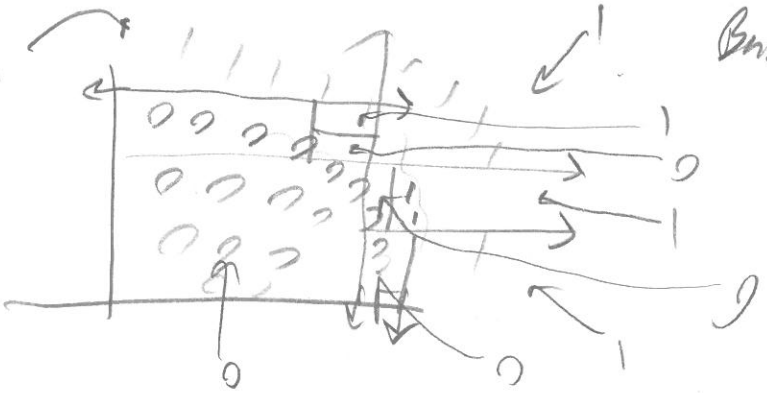


$$\mathcal{H} = \{ \mathbb{1}_{w_0 + w_1 x_1 + w_2 x_2 + w_3 x_1^2 + w_4 x_2^2 \geq 0} \;\; ; \vec{w} \in \mathbb{R}^5 \}$$

$$A: \operatorname{argmin} \{ \frac{1}{n} \sum_{i=1}^{n} \max\{0, \frac{1}{2} \cdot (y_i - \frac{1}{2})(\vec{w} \cdot \vec{x}_i - b)\} + \lambda \|\vec{w}\|^2 \}$$

Did we ever discuss general classification into $K > 2$ labels? No.

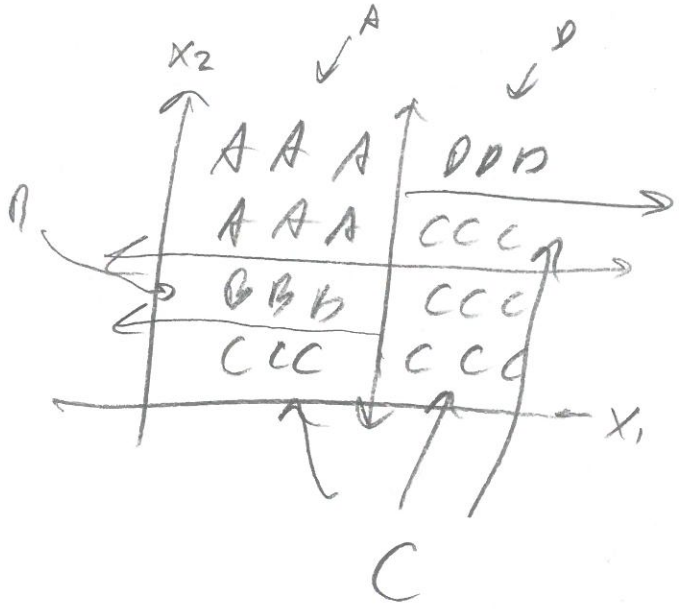
How do trees work?



Basically the same!!

Can it do more than $K=2$? Yes... very powerful..

Perceptron / SVM can do it (as of now).



Classification Tree Algorithm

(1) Begin with all training data

(2) For every possible split, calculate the Gini "impurity" measures:

$$\text{Gini}_L := \sum_{\ell=1}^{K} \hat{p}_\ell (1-\hat{p}_\ell), \quad \text{Gini}_R := \sum_{\ell=1}^{K} \hat{p}_\ell (1-\hat{p}_\ell)$$

where $\hat{p}_\ell = \dfrac{\#\, y_i \text{ in category } \ell}{n, \#\text{ in node}}$

(3) Find the split with the lowest weighted avg of Gini impurity measures:

$$\text{Gini}_{\text{avg}} := \frac{n_L \text{Gini}_L + n_R \text{Gini}_R}{n_L + n_R}$$

(4) Create the split and split the data in the node correctly among the two new daughter nodes

(5) Repeat steps 2-4 until "stop" (node has $\le N_0$ observations) Default: $N_0 = 1$!!!

(6) For all leaf nodes, assign $\hat{y} = \text{Mode}[\vec{y}_0]$ where $\vec{y}_0$ are the obs. of the y_i's in the leaf node. DEMO