Decision Trees

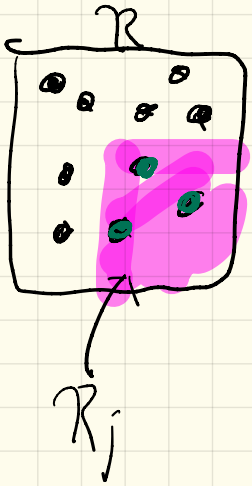
Purity

purity of a region R_j
fraction of the mixture of classes of pts in R_j
w/ majority label

let D_j be the pts in D w/in R_j

$n_j = |D_j|$
n_{ji} = # of pts in D_j w/ label i

$$\text{purity}(D_j) = \max_i \left\{ \frac{n_{ji}}{n_j} \right\}$$



Decision Tree Algo

Give dataset D
leaf size $\eta \leftarrow$ (eta)
purity threshold π

Recursively:
pick "best" split point (X_j, v) *
partition data in D_Y, D_N
recurse (D_Y)
recurse (D_N)

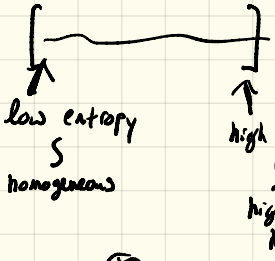
Stop:
- # of pts in partition is below η (avoid overfitting)
or - purity is above π

Algorithm 19.1: Decision Tree Algorithm

```
DECISIONTREE (D, η, π):
n ← |D| // partition size
n_i ← |{x_j | x_j ∈ D, y_j = c_i}| // size of class c_i
purity(D) ← max_i {n_i / n}
if n ≤ η or purity(D) ≥ π then // stopping condition
    c* ← arg max_{c_i} {n_i / n} // majority class
    create leaf node, and label it with class c*
    return
(split point*, score*) ← (∅, 0) // initialize best split point
foreach (attribute X_j) do
    if (X_j is numeric) then
        (v, score) ← EVALUATE-NUMERIC-ATTRIBUTE(D, X_j)
        if score > score* then (split point*, score*) ← (X_j ≤ v, score)
    else if (X_j is categorical) then
        (V, score) ← EVALUATE-CATEGORICAL-ATTRIBUTE(D, X_j)
        if score > score* then (split point*, score*) ← (X_j ∈ V, score)
// partition D into D_Y and D_N using split point*, and call
   recursively
D_Y ← {x^T | x ∈ D satisfies split point*}
D_N ← {x^T | x ∈ D does not satisfy split point*}
create internal node split point*, with two child nodes, D_Y and D_N
DECISIONTREE(D_Y); DECISIONTREE(D_N)
```

Split point evaluation measures:

Entropy – amount of disorder in a system



$$H(D) = -\sum_{i=1}^{k} P(c_i|D) \log P(c_i|D)$$

- k: # of class labels in D
- $P(c_i|D)$: prob of a class C_i in dataset D
- $P(c_i|D)$: $\frac{\#\text{ of records w/ class label } c_i}{\#\text{ of record in } D}$

$P(c_\ell|D) = 1$

$\Rightarrow \forall j = \ell \quad P(c_j|D) = 0$

$\Rightarrow H(D) = -[$

~~$0 * \log t$~~ $\quad i=1$

~~$0 *$~~ $\quad i=2$

$\vdots$

$1 \cdot \log(1) \quad i=\ell$

$\vdots$

$0 *$ $\quad i$

$] = 0 \Rightarrow$ no entropy

$P(c_i|D) = \frac{1}{k} \quad \forall i$

$\Rightarrow H(D) = -\sum_{i=1}^{k} \frac{1}{k} \log\left(\frac{1}{k}\right)$

$= -\frac{1}{k} \sum_{i=1}^{k} \log\left(\frac{1}{k}\right)$

$= -\frac{1}{k} \log \frac{1}{k}$

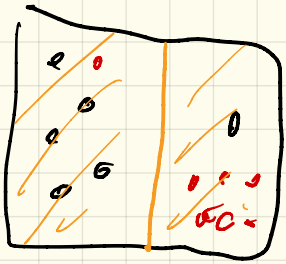
$= -\frac{1}{k} \log k^{(-k)}$

$= \log k$

$= \log(k) \leftarrow$ max entropy

Let D_Y, D_N be partitions of D

$n = |D|$ $n_Y = |D_Y|$ $n_N = |D_N|$



split entropy: weighted entropy of each partion

$$H(D_Y, D_N) = \frac{n_Y}{n} H(D_Y) + \frac{n_N}{n} H(D_N)$$

information gain of a split point as decrease in entropy

$$Gain(D, D_Y, D_N) = H(D) - H(D_Y, D_N)$$

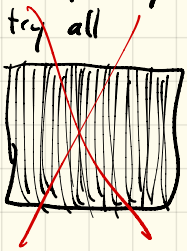
⇒ high information gain means better split point
pick split point w/ highest information gain

Eval split pts

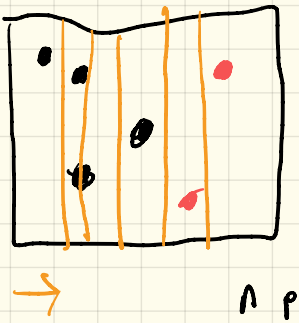
Alg ❶ for finging split pts

compute D_Y D_N and $P(c_i | D_Y)$, $P(c_i, D_N)$, $Gain(D, D_Y, D_N)$

try all



Alg 1



→

n pts

⇒ n-1 places to consider splitting