Machine Learning

- based on observations x
- learn the function $\hat{y} = f(\mathsf{x}) \approx \mathsf{y}$

Machine Learning

- based on observations $\mathbf{x}$
- learn the function $\hat{y} = f(\mathbf{x}) \approx \mathrm{y}$

How to learn ?

- Iterative process
- Take a bunch of x
- Compute a frontier to separate x
- Compare on which side x are with where they should be
- Propagate the error

Machine Learning

- based on observations $\mathbf{x}$
- learn the function $\hat{y} = f(\mathbf{x}) \approx \mathrm{y}$

How to learn ?

- Iterative process
- Take a bunch of x
- Compute a frontier to separate x
- Compare on which side x are with where they should be
- Propagate the error

Overfitting

- If the learning process keeps going...
- No errors on seen data
- But what about other x (unseen in the training phase)

Diff between Decision and Regression

- Decision = predict a label
- Regression = predict a value

Diff between Decision and Regression

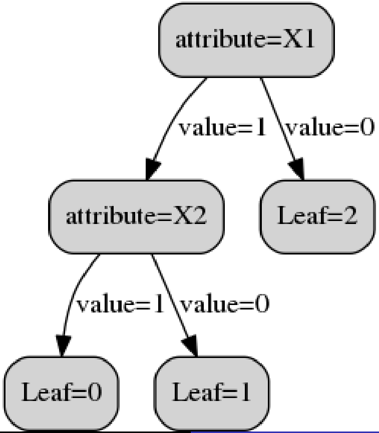
- Decision = predict a label
- Regression = predict a value

Regression Trees

- Similar to Decision Trees
- Problem is more complex

Decision Trees

- Frontier is based on decision rules
- Model has a tree hierarchy
- Each level is a test on attributes



How to choose the order of attributes ?

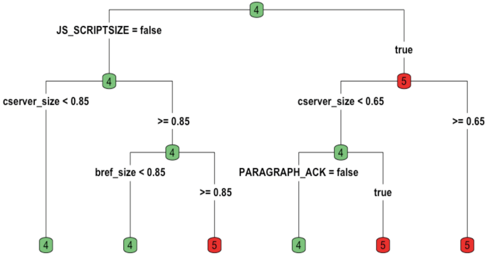
- Multiple possible heuristics
- Most popular : Information gain ; gini

How to choose the order of attributes ?

- Multiple possible heuristics
- Most popular : Information gain ; gini

Information gain

- Entropy based heuristic
- What is the attribute that discriminate most ?



Coming back at overfitting

- Few prediction errors on the training set
- Much more on the test set

Coming back at overfitting

- Few prediction errors on the training set
- Much more on the test set

How to avoid it ?

- Constrain the growth of the tree
- Prune the tree

Coming back at overfitting

- Few prediction errors on the training set
- Much more on the test set

How to avoid it ?

- Constrain the growth of the tree
- Prune the tree

Pruning

- based on heuristic
- Impurity measure
- Trade-off between the depth of the tree and number of errors

```
Input: attributes, examples, selection_heuristic, prunning_heuristic, strucutre_constraints
Output: tree : a decision tree
$tree = \emptyset$
$attribute = choose\_best(attributes, examples, heuristic)$
$tree = create_root(attribute)$
$attributes = attributes - attribute$
while $attributes <> \emptyset \| strucutre\_constraintsmet$ do
    $attribute = choose\_best(attributes, examples, heuristic)$
    $tree = add_child(attribute)$
    $attributes = attributes - attribute$
end while
$tree = prune(tree, prunning\_heuristic)$
```

Exploitation

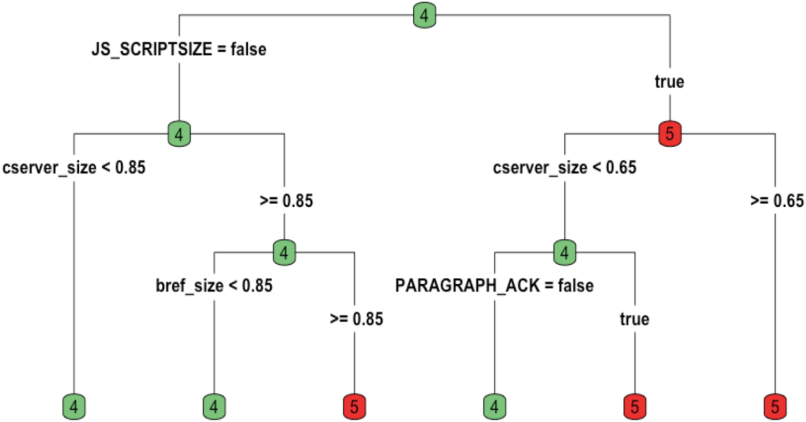
- For each incoming new data
- Begin at the root of the tree
- Follow path depending on results of tests
- The ending leaf gives the class

Exploitation

- For each incoming new data
- Begin at the root of the tree
- Follow path depending on results of tests
- The ending leaf gives the class

Decision rules

- New constraints can be extracted
- Simply follow the path !


4
JS_SCRIPTSIZE = false
true
4
cserver_size < 0.85
>= 0.85
4
bref_size < 0.85
>= 0.85
5
cserver_size < 0.65
>= 0.65
4
PARAGRAPH_ACK = false
true
4
4
5
4
5
5