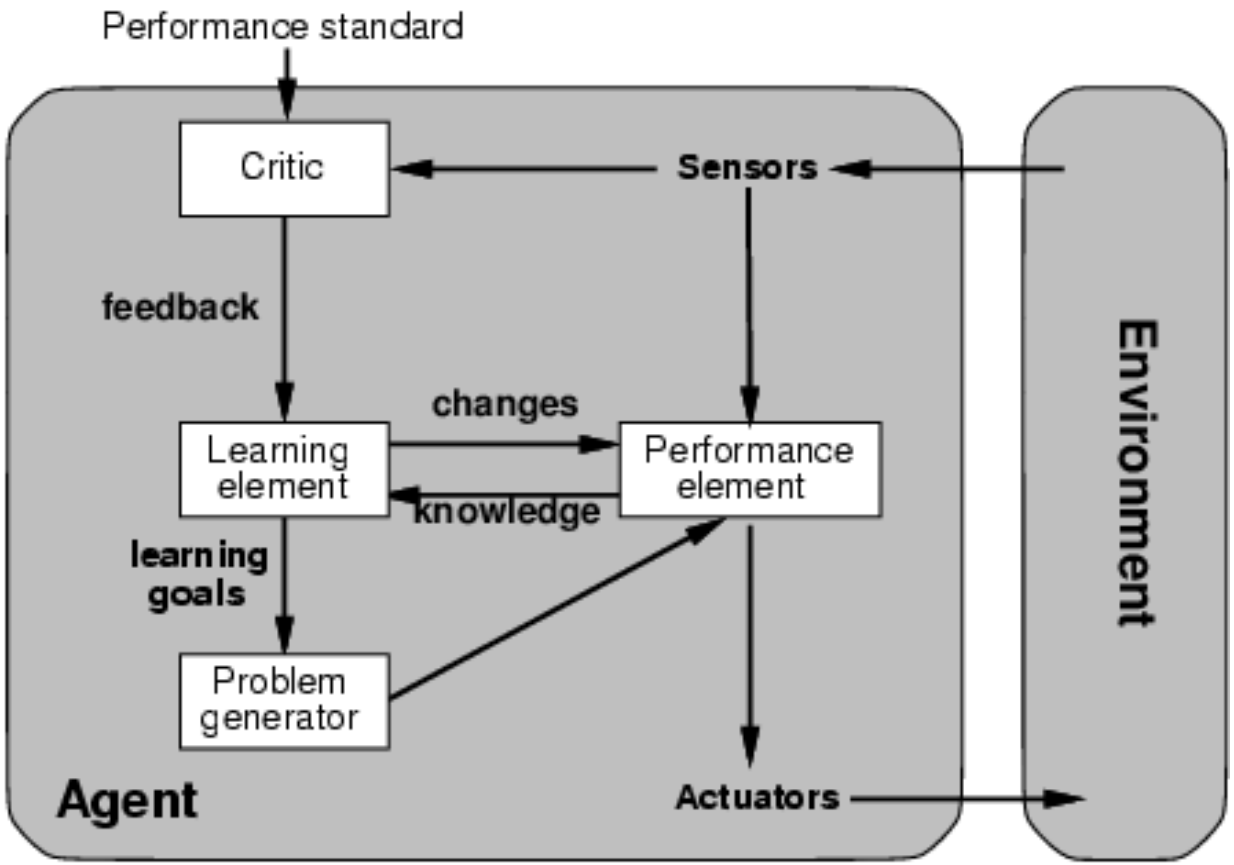
Machine Learning



Learning Schemes

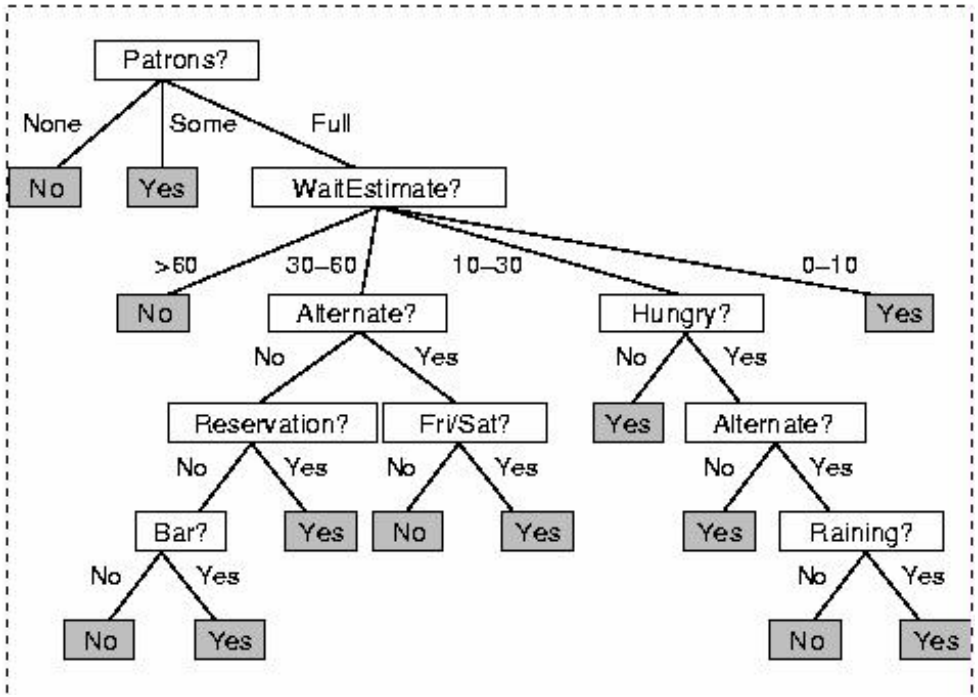
- supervised learning
  - learns from examples
- reinforcement learning
  - learns how to act based on observations
- unsupervised learning
  - groups data into categories; no labeled examples available

Supervised Learning

- given examples (x, $f(x)$), learn (x, $h(x)$) where $h(x) \approx f(x)$
  - regression: $h(x)$ is a continuous value
  - classification: $h(x)$ is a class label
- Learning can be:
  - Incremental: update hypothesis when new example seen
  - Batch: hypotheses generated after considering a training set of n examples

Simple Supervised Learning:
Decision Trees

- examples (training set) described by:
  - input: the values of attributes
  - output: the classification (yes/no)
- can represent any Boolean function



Example	Attributes										Goal
	Alt	*Bar*	*Fri*	*Hun*	*Pat*	*Price*	*Rain*	*Res*	*Type*	*Est*	*WillWait*
X_1	*Yes*	*No*	*No*	*Yes*	*Some*	$$$	*No*	*Yes*	*French*	*0–10*	*Yes*
X_2	*Yes*	*No*	*No*	*Yes*	*Full*	$	*No*	*No*	*Thai*	*30–60*	*No*
X_3	*No*	*Yes*	*No*	*No*	*Some*	$	*No*	*No*	*Burger*	*0–10*	*Yes*
X_4	*Yes*	*No*	*Yes*	*Yes*	*Full*	$	*No*	*No*	*Thai*	*10–30*	*Yes*
X_5	*Yes*	*No*	*Yes*	*No*	*Full*	$$$	*No*	*Yes*	*French*	*>60*	*No*
X_6	*No*	*Yes*	*No*	*Yes*	*Some*	$$	*Yes*	*Yes*	*Italian*	*0–10*	*Yes*
X_7	*No*	*Yes*	*No*	*No*	*None*	$	*Yes*	*No*	*Burger*	*0–10*	*No*
X_8	*No*	*No*	*No*	*Yes*	*Some*	$$	*Yes*	*Yes*	*Thai*	*0–10*	*Yes*
X_9	*No*	*Yes*	*Yes*	*No*	*Full*	$	*Yes*	*No*	*Burger*	*>60*	*No*
X_{10}	*Yes*	*Yes*	*Yes*	*Yes*	*Full*	$$$	*No*	*Yes*	*Italian*	*10–30*	*No*
X_{11}	*No*	*No*	*No*	*No*	*None*	$	*No*	*No*	*Thai*	*0–10*	*No*
X_{12}	*Yes*	*Yes*	*Yes*	*Yes*	*Full*	$	*No*	*No*	*Burger*	*30–60*	*Yes*

Inducing Decision Trees

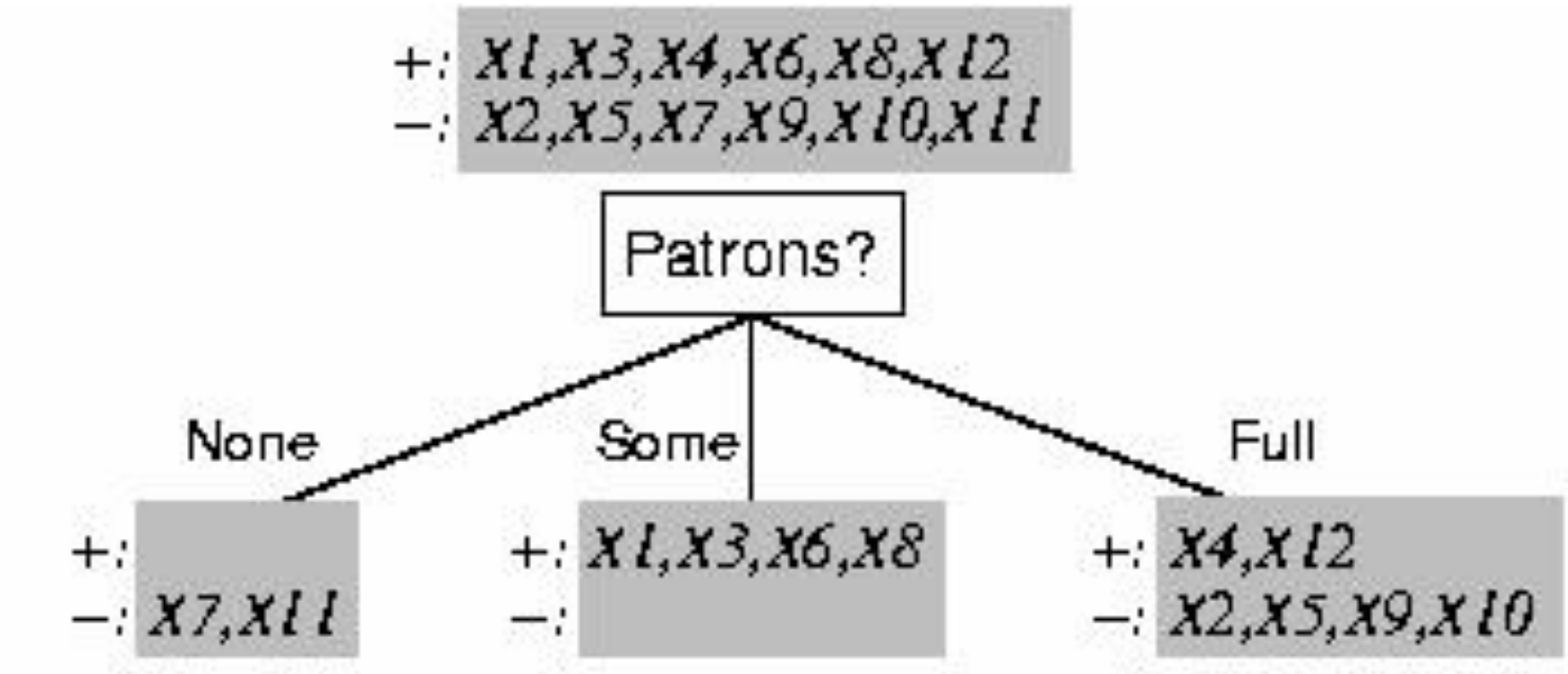
- option #1: build full look-up table
  - one path for each training example
  - going down path tests each attribute in turn
  - leaf assigned the classification of example
- problems:
  - does not learn patterns
  - size is 2^n—hardly the most concise description
  - other problems -- we'll see this soon

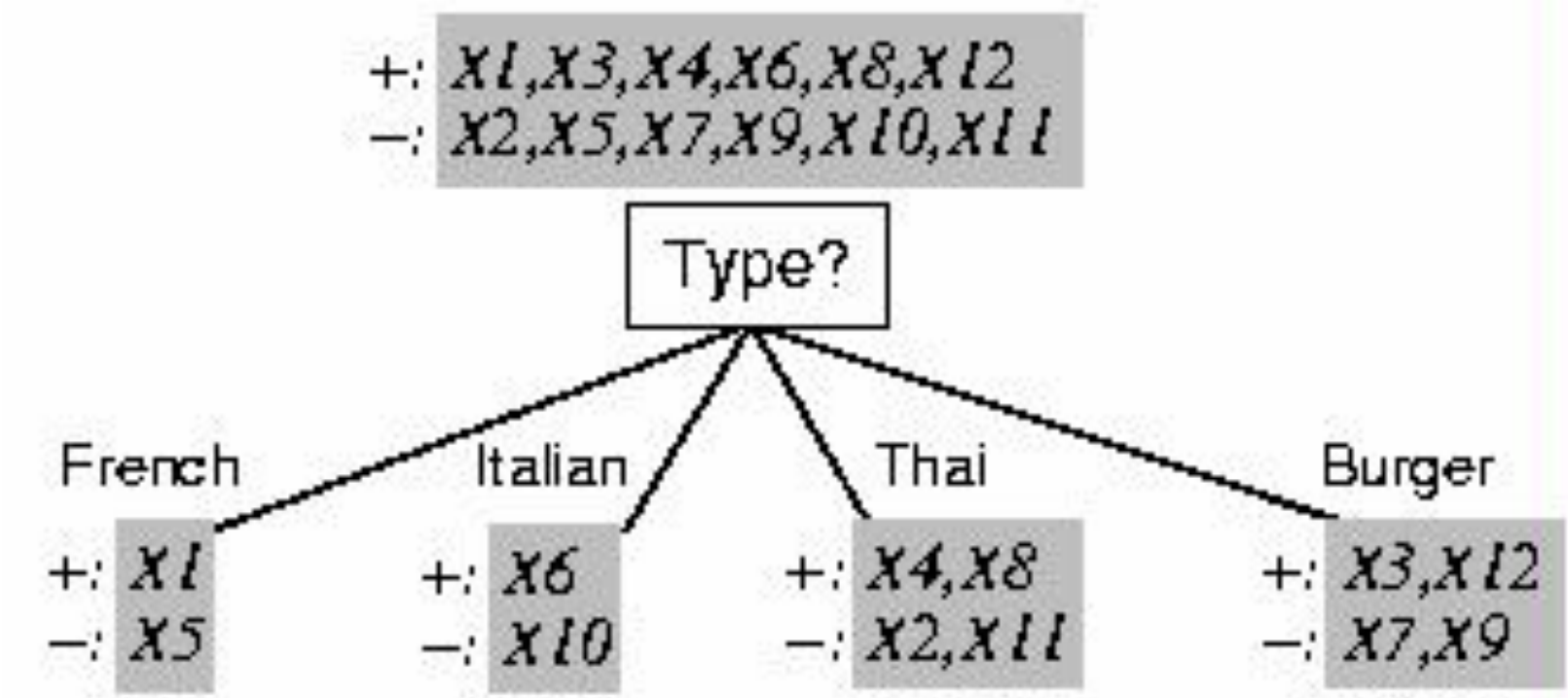
Fixing the 2^n problem

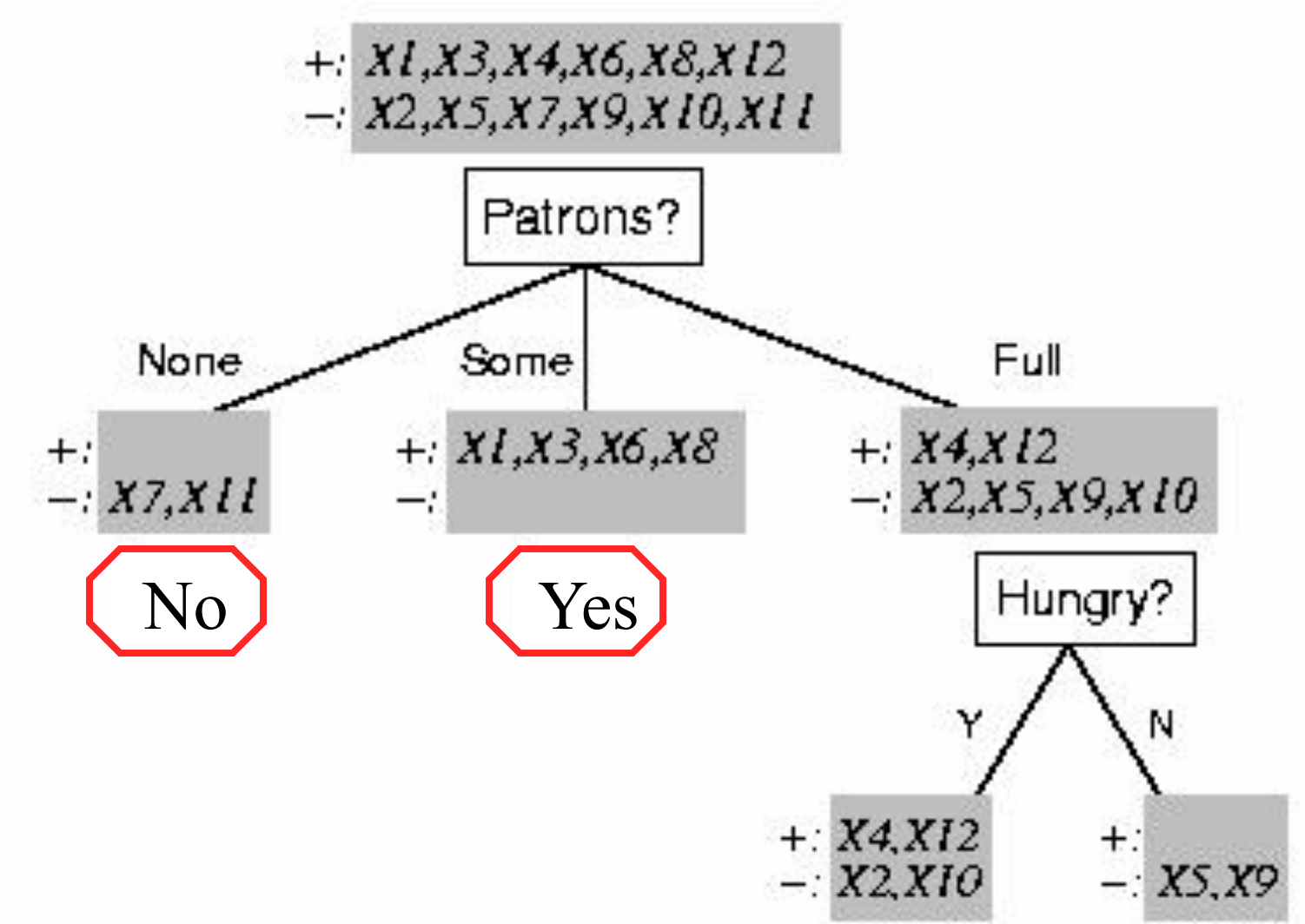
- form more compact representation
  - look for redundancy of attributes
  - but not always possible, e.g.,
    - parity function
    - majority function

Recursive Decision Tree Induction

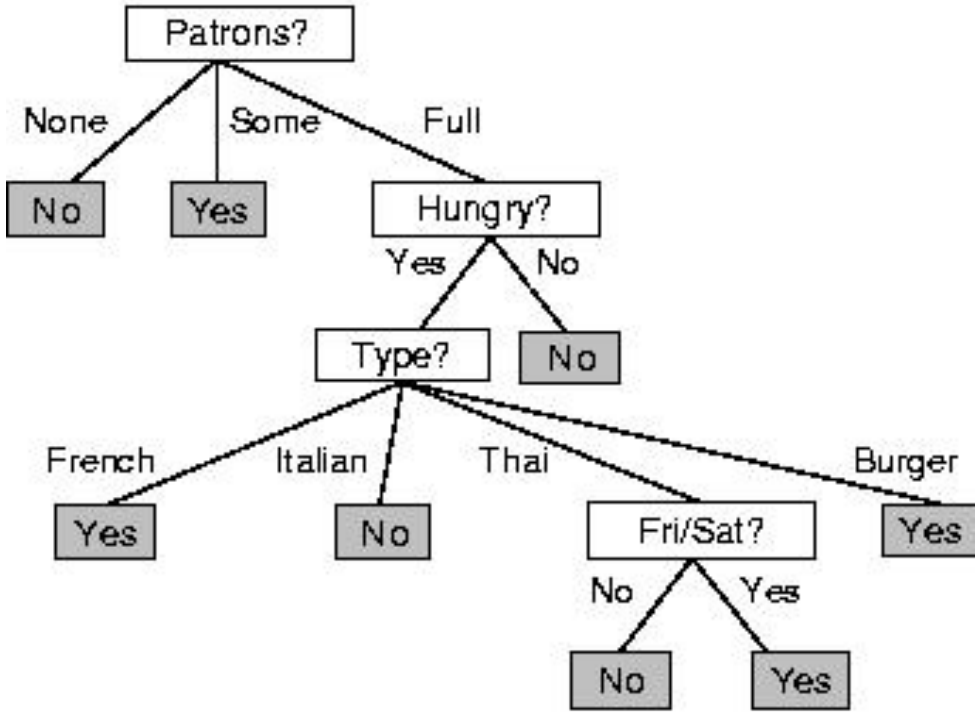
- **Case 1:** if we have some +, some - examples, split based on *most important* attribute
  - defined by information theoretic measures
- **Case 2:** if remaining examples all + (or -) then done
  - can answer yes or no
- **Case 3:** if there is some value for a particular attribute but no examples available to classify it
  - use default value (majority classification) from node's parent
- **Case 4:** if no attributes left, but we still have some +, some -, examples, problem:
  - data noise, insufficient information, or nondeterministic

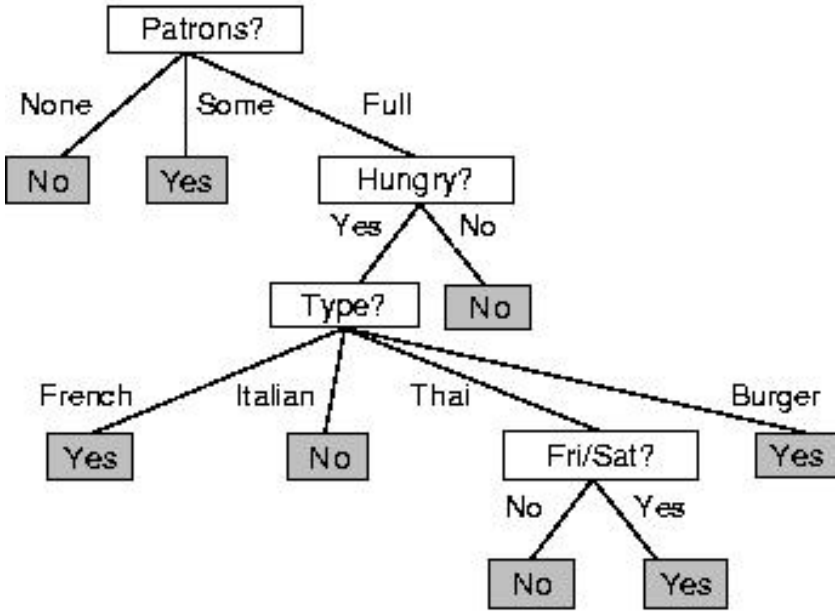

+: x1,x3,x4,x6,x8,x12
−: x2,x5,x7,x9,x10,x11
Patrons?
None
Some
Full
+:
−: x7,x11
+: x1,x3,x6,x8
−:
+: x4,x12
−: x2,x5,x9,x10


+: X1,X3,X4,X6,X8,X12
−: X2,X5,X7,X9,X10,X11
Type?
French
Italian
Thai
Burger
+: X1
−: X5
+: X6
−: X10
+: X4,X8
−: X2,X11
+: X3,X12
−: X7,X9

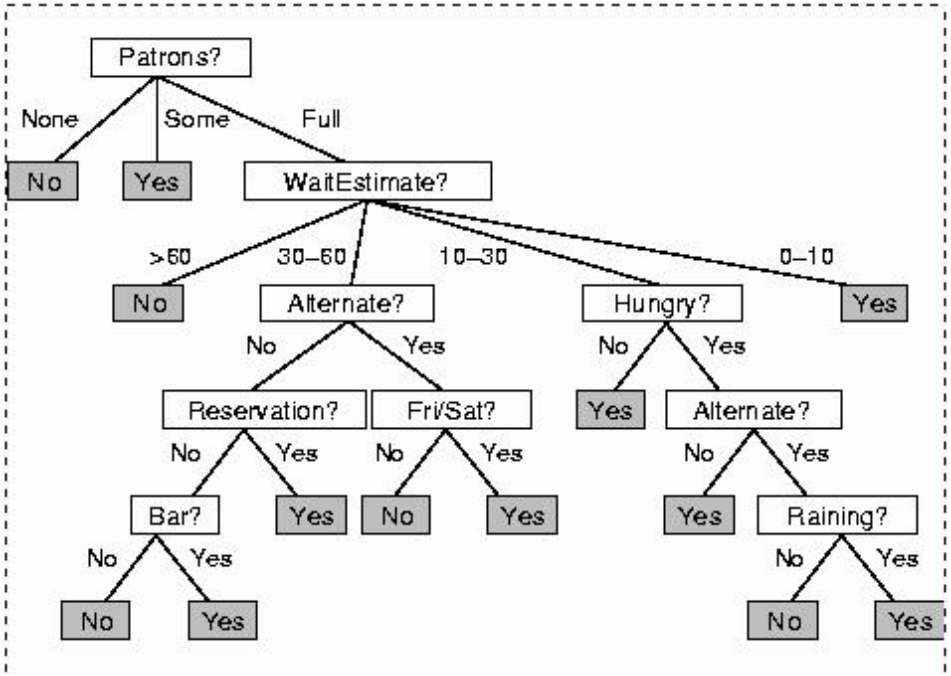

+: X1,X3,X4,X6,X8,X12
–: X2,X5,X7,X9,X10,X11
Patrons?
None
Some
Full
+:
–: X7,X11
+: X1,X3,X6,X8
–:
+: X4,X12
–: X2,X5,X9,X10
No
Yes
Hungry?
Y
N
+: X4,X12
–: X2,X10
+:
–: X5,X9

Example	Attributes										Goal
	Alt	*Bar*	*Fri*	*Hun*	*Pat*	*Price*	*Rain*	*Res*	*Type*	*Est*	*WillWait*
X_1	*Yes*	*No*	*No*	*Yes*	*Some*	$$$	*No*	*Yes*	*French*	*0–10*	*Yes*
X_2	*Yes*	*No*	*No*	*Yes*	*Full*	$	*No*	*No*	*Thai*	*30–60*	*No*
X_3	*No*	*Yes*	*No*	*No*	*Some*	$	*No*	*No*	*Burger*	*0–10*	*Yes*
X_4	*Yes*	*No*	*Yes*	*Yes*	*Full*	$	*No*	*No*	*Thai*	*10–30*	*Yes*
X_5	*Yes*	*No*	*Yes*	*No*	*Full*	$$$	*No*	*Yes*	*French*	*>60*	*No*
X_6	*No*	*Yes*	*No*	*Yes*	*Some*	$$	*Yes*	*Yes*	*Italian*	*0–10*	*Yes*
X_7	*No*	*Yes*	*No*	*No*	*None*	$	*Yes*	*No*	*Burger*	*0–10*	*No*
X_8	*No*	*No*	*No*	*Yes*	*Some*	$$	*Yes*	*Yes*	*Thai*	*0–10*	*Yes*
X_9	*No*	*Yes*	*Yes*	*No*	*Full*	$	*Yes*	*No*	*Burger*	*>60*	*No*
X_{10}	*Yes*	*Yes*	*Yes*	*Yes*	*Full*	$$$	*No*	*Yes*	*Italian*	*10–30*	*No*
X_{11}	*No*	*No*	*No*	*No*	*None*	$	*No*	*No*	*Thai*	*0–10*	*No*
X_{12}	*Yes*	*Yes*	*Yes*	*Yes*	*Full*	$	*No*	*No*	*Burger*	*30–60*	*Yes*





Induced tree



Original tree

- induced decision tree looks nothing like the original
  - it's much smaller, but is it wrong?
- doesn't necessarily learn original function
  - ignores *Raining* and *Reservation*
- what happens if *wait = 5 minutes*,*restaurant is full*?
  - induced tree says not to wait if not hungry