Data Mining and Data Warehousing

Chapter 4

Classification and Prediction

Instructor: Suresh Pokharel
ME in ICT (Asian Institute of Technology, Thailand)
BE in Computer (NCIT, Pokhara University)

What is classification?

- is a data mining technique used to predict the category of categorical data by building a model based on some predictor variables (to classify data).
- Predictor variable/attribute is called class label attribute (predefined class)

What is classification?

It is a two-step process

1. Model Construction (learning step or training phase)

   - build a model to explain the target concept

   - model is represented as classification rules, decision trees, or mathematical formulae.

2. Model Usage

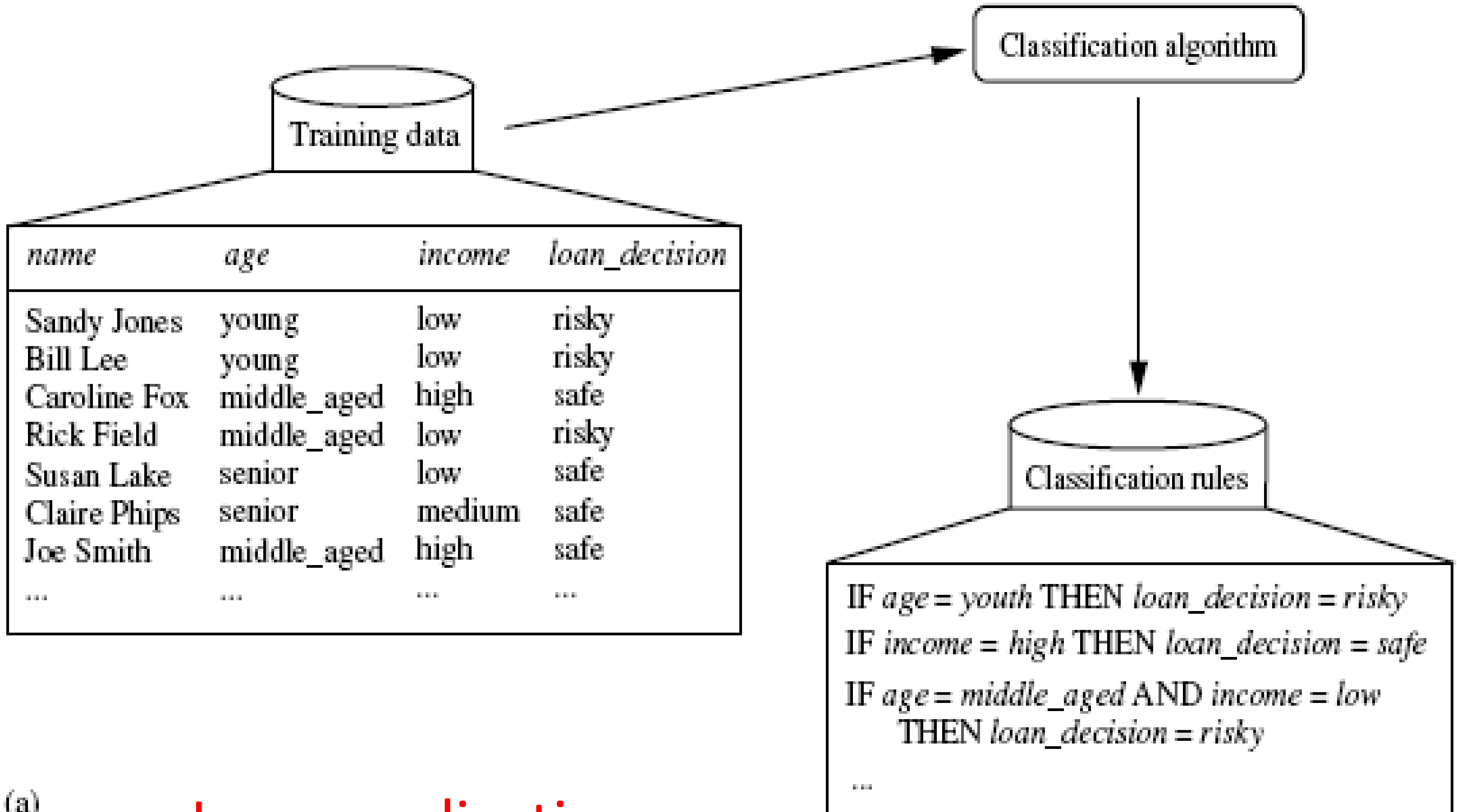
   - is used for classifying future or unknown cases

   - estimate the accuracy of the model

Example

RID	*age*	*income*	*student*	*credit_rating*	*Class: buys_computer*
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
3	middle_aged	high	no	fair	yes
4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	excellent	no
7	middle_aged	low	yes	excellent	yes
8	youth	medium	no	fair	no
9	youth	low	yes	fair	yes
10	senior	medium	yes	fair	yes
11	youth	medium	yes	excellent	yes
12	middle_aged	medium	no	excellent	yes
13	middle_aged	high	yes	fair	yes
14	senior	medium	no	excellent	no

Step 1

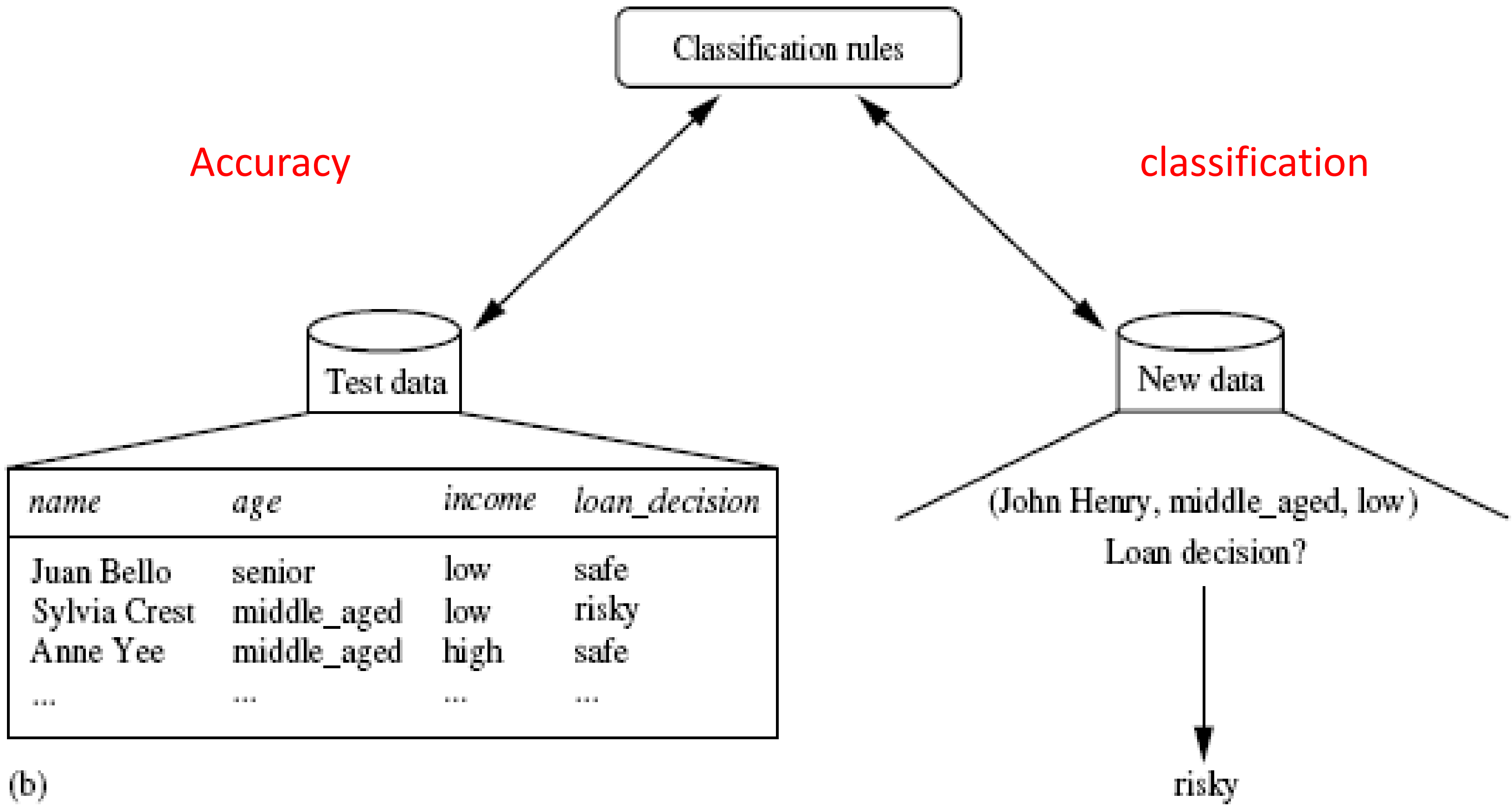


name	*age*	*income*	*loan_decision*
Sandy Jones	young	low	risky
Bill Lee	young	low	risky
Caroline Fox	middle_aged	high	safe
Rick Field	middle_aged	low	risky
Susan Lake	senior	low	safe
Claire Phips	senior	medium	safe
Joe Smith	middle_aged	high	safe
...	...	...	...

(a)

Loan application

Step 2

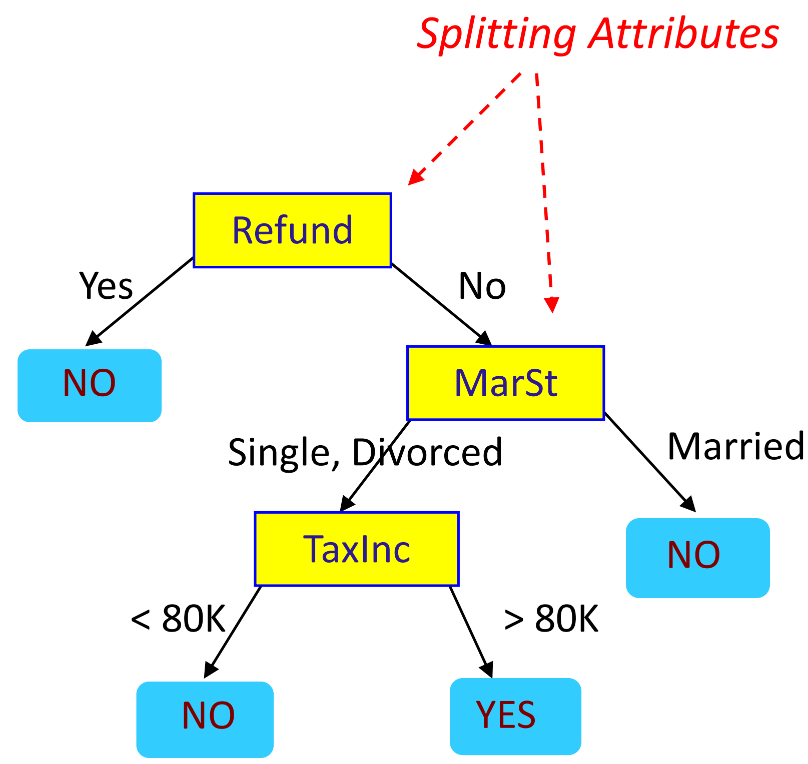

Classification rules
Accuracy
classification
Test data
New data
name | age | income | loan_decision
Juan Bello | senior | low | safe
Sylvia Crest | middle_aged | low | risky
Anne Yee | middle_aged | high | safe
... | ... | ... | ...
(John Henry, middle_aged, low)
Loan decision?
risky
(b)

Example of a Decision Tree

categorical categorical continuous class

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Training Data

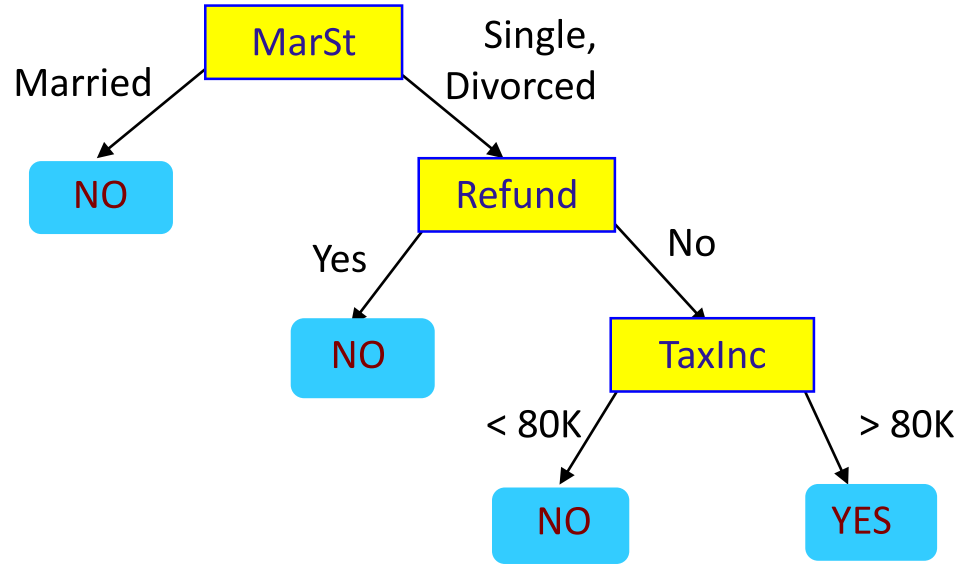


Model: Decision Tree

Another Example of Decision Tree

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

categorical, categorical, continuous, class

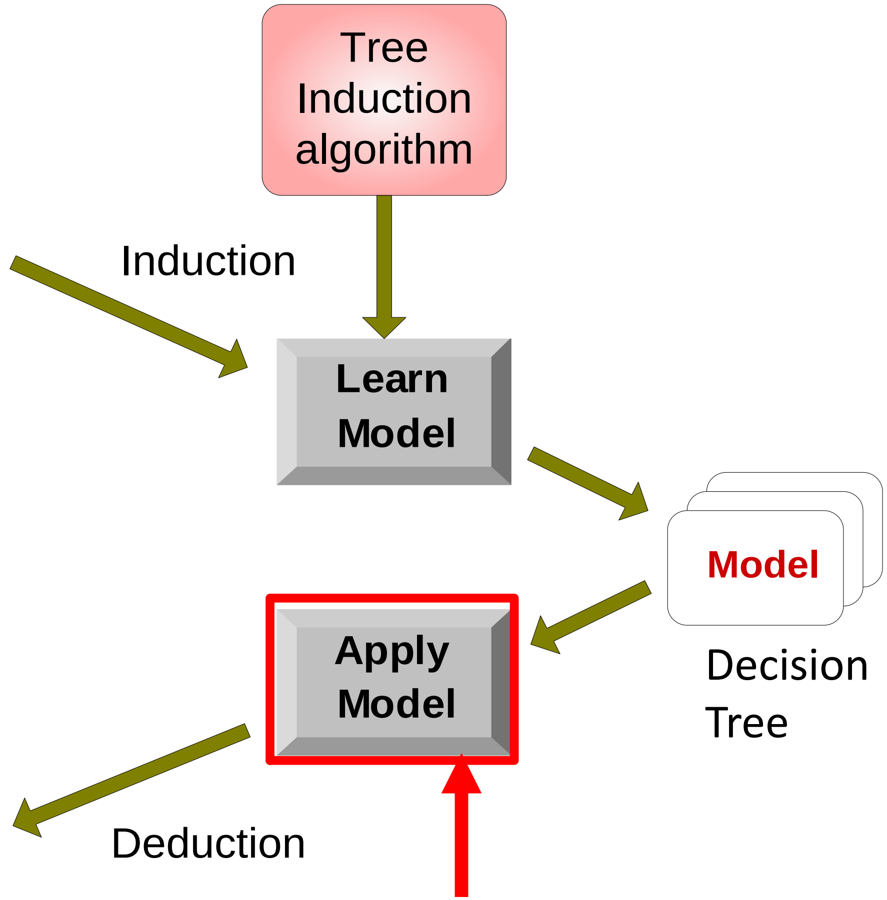


There could be more than one tree that fits the same data!

Decision Tree Classification Task

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

Training Set



Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

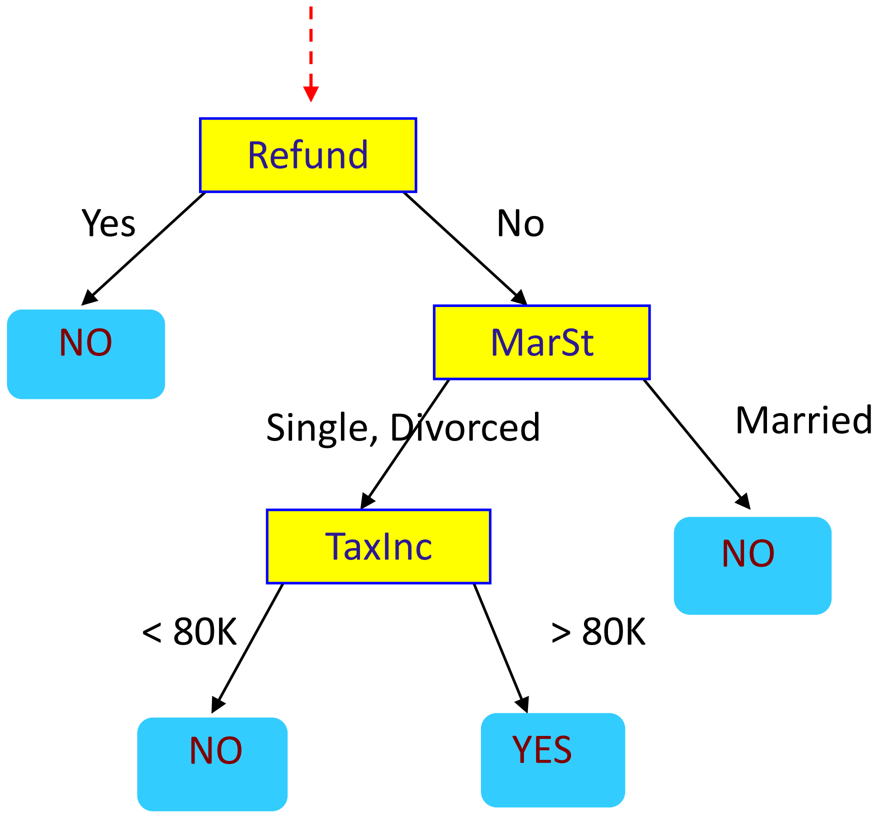
Test Set

Apply Model to Test Data

Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

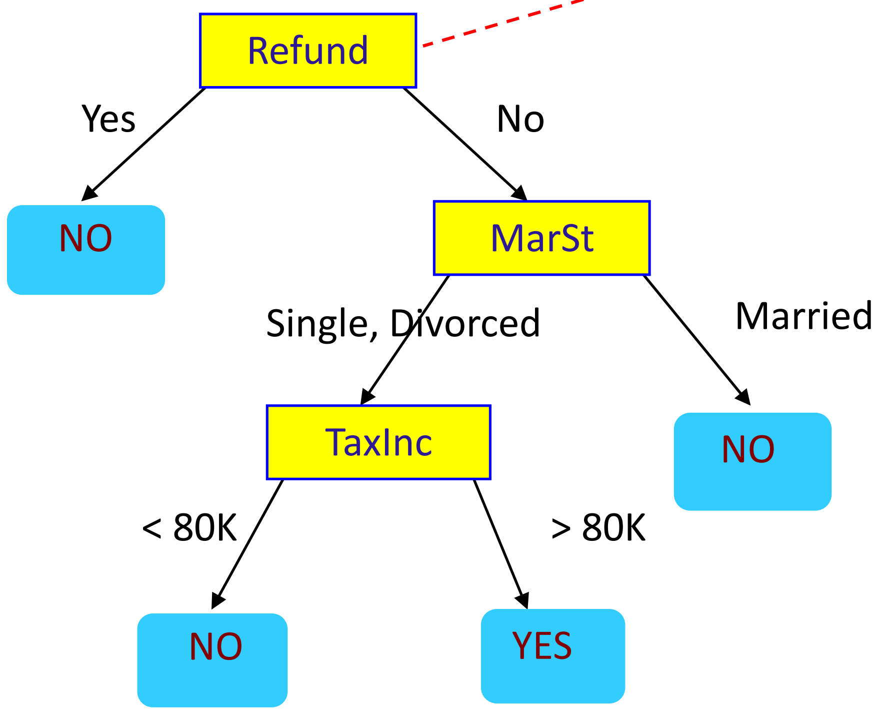
Start from the root of tree.



Apply Model to Test Data

Test Data

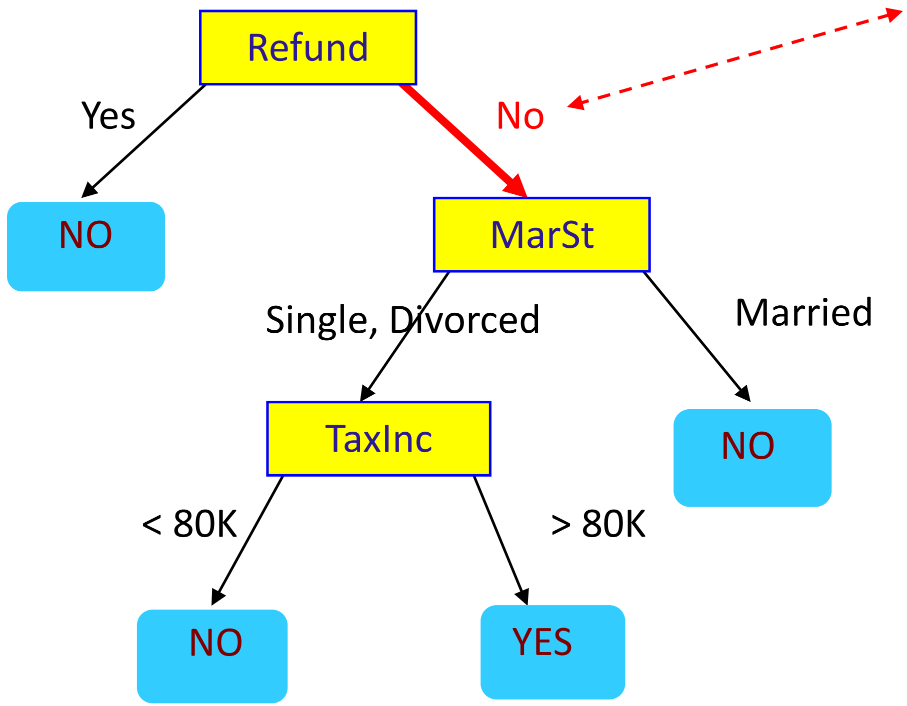
Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?



Apply Model to Test Data

Test Data

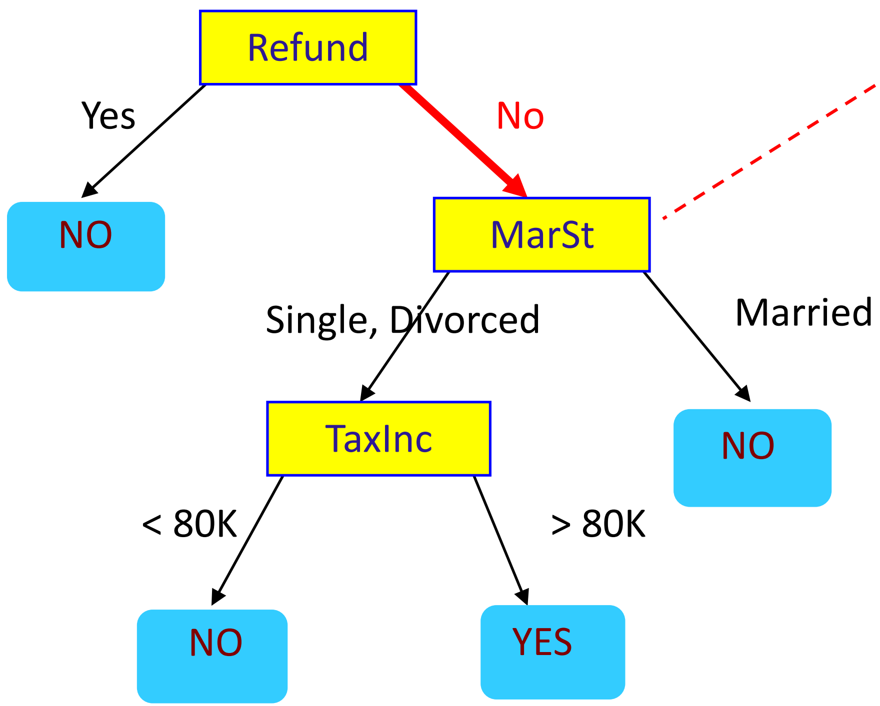
Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?



Apply Model to Test Data

Test Data

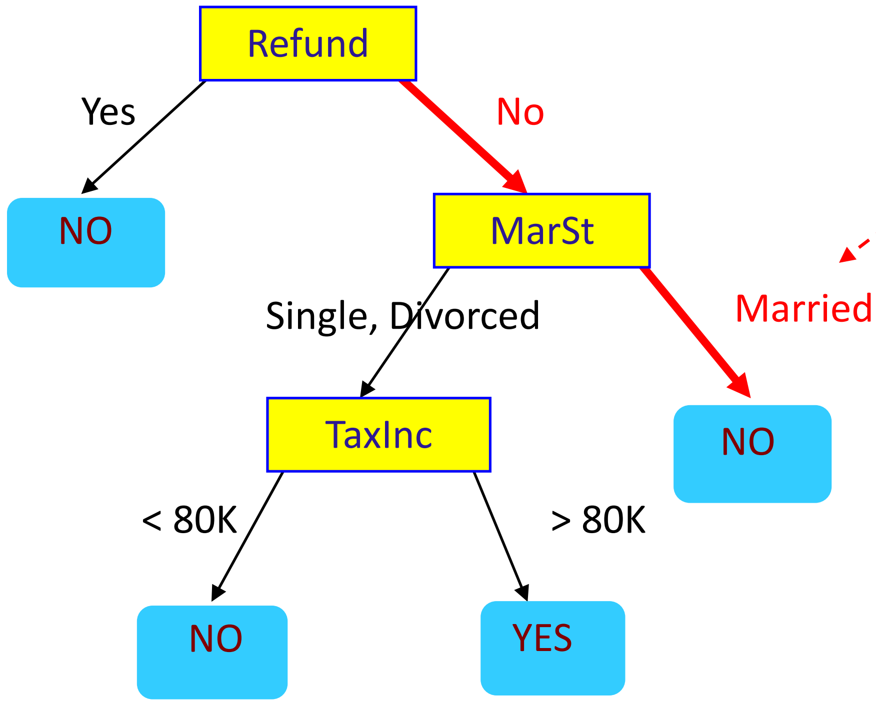
Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?



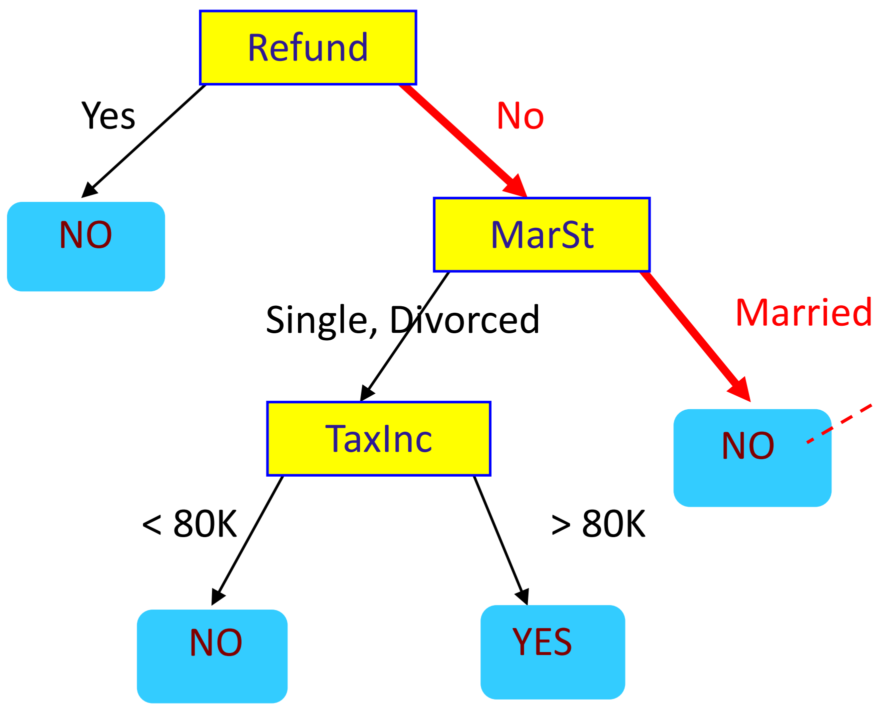
Apply Model to Test Data

Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?



Apply Model to Test Data



Test Data

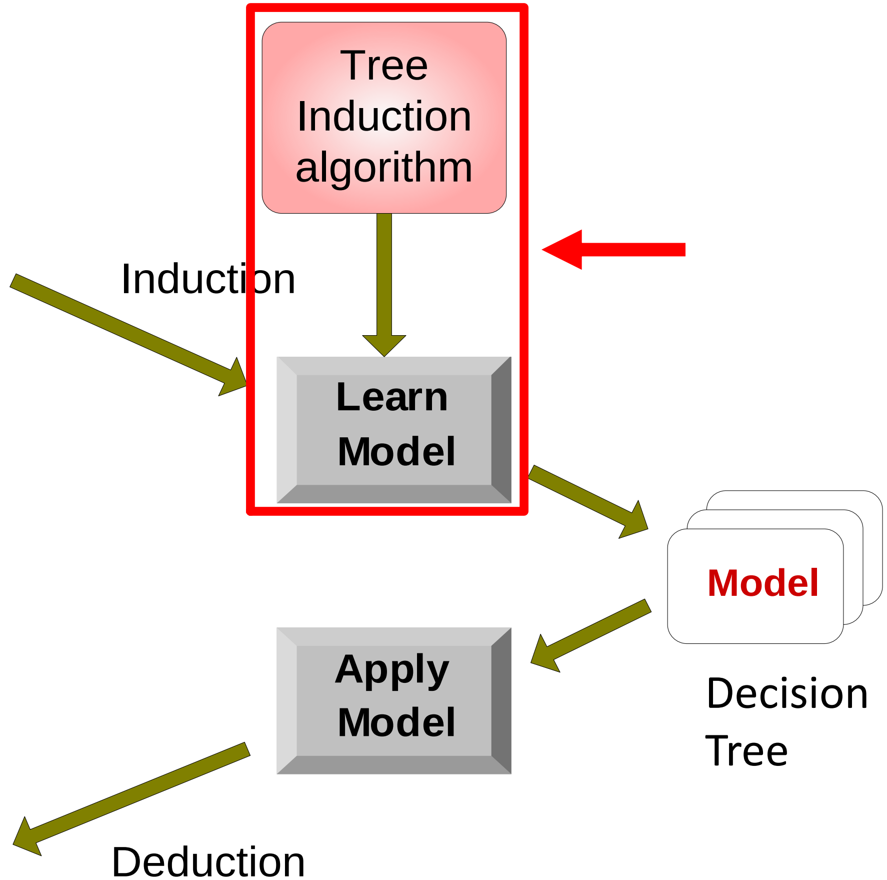
Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Assign Cheat to "No"

Decision Tree Classification Task

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

Training Set



Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set

What is Prediction?

- Models continuous-valued functions, i.e. predicts unknown or missing values (numeric)
- Lost terminology of "class label attribute", instead we use "predicted attribute"
- Viewed as a mapping or function $y = f(X)$
- Example: predict the amount (in dollars) that would be safe for the bank to loan an application

Supervised & Unsupervised Learning

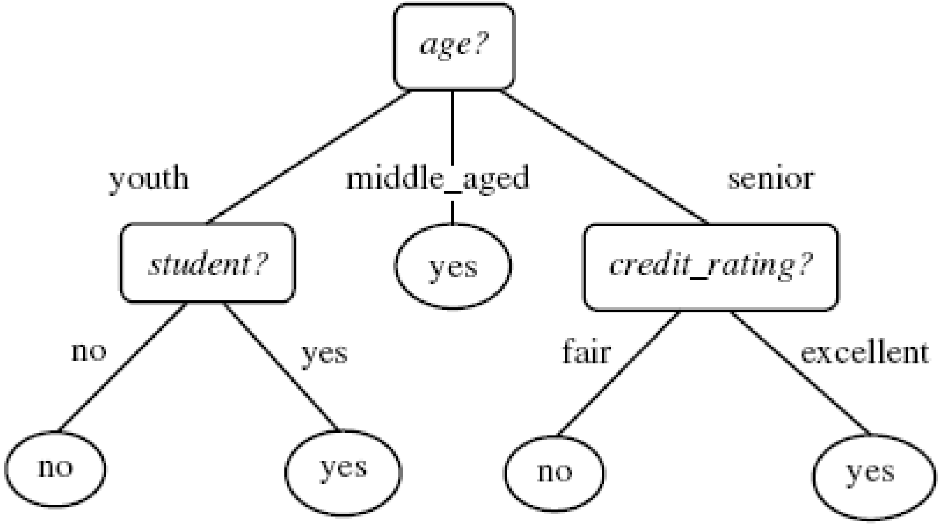
- Supervised Learning (Classification)
  - Supervision: The training data are accompanied by labels indicating the class of the observations
  - New data is classified based on the training set
- Unsupervised Learning (Clustering)
  - The class labels of training data is unknown
  - Given a set of measurements, observations, etc. with the aim of establishing the existence of classes or clusters in the data

Classification by decision tree induction

What is decision tree?

- flow chart like tree structure
- internal node denotes a test on an attribute
- each branch represents an outcome of the test
- each leaf node holds a class label

Decision tree for the concept buys computer



Why decision tree?

- Construction does not require any domain knowledge
- Can handle high dimensional data
- Learning step is simple and fast
- In general have good accuracy
- People are able to understand decision tree models after a brief explanation

Attribute selection measures

Table 6.1 Class-labeled training tuples from the *AllElectronics* customer database.

RID	*age*	*income*	*student*	*credit_rating*	*Class: buys_computer*
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
3	middle_aged	high	no	fair	yes
4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	excellent	no
7	middle_aged	low	yes	excellent	yes
8	youth	medium	no	fair	no
9	youth	low	yes	fair	yes
10	senior	medium	yes	fair	yes
11	youth	medium	yes	excellent	yes
12	middle_aged	medium	no	excellent	yes
13	middle_aged	high	yes	fair	yes
14	senior	medium	no	excellent	no

Information Gain, gain ratio and gini index

Attribute selection measures

- Information Gain

the expected information needed to classify a tuple in *D : (*entropy of *D)*

$$Info(D) = -\sum_{i=1}^{m} p_i \log_2(p_i),$$

- ➢ *Pi = |C*
- ➢ *A* having *v* distinct values, $\{a_1, a_2, \ldots, a_v\}$
- ➢ $D_1, D_2, \ldots, D_v$

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times Info(D_j).$$

Attribute selection measures

$$Gain(A) = Info(D) - Info_A(D).$$

(Choose maximum value)

Table 6.1 Class-labeled training tuples from the *AllElectronics* customer database.

RID	*age*	*income*	*student*	*credit_rating*	*Class: buys_computer*
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
3	middle_aged	high	no	fair	yes
4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	excellent	no
7	middle_aged	low	yes	excellent	yes
8	youth	medium	no	fair	no
9	youth	low	yes	fair	yes
10	senior	medium	yes	fair	yes
11	youth	medium	yes	excellent	yes
12	middle_aged	medium	no	excellent	yes
13	middle_aged	high	yes	fair	yes
14	senior	medium	no	excellent	no

Example A is discrete-valued

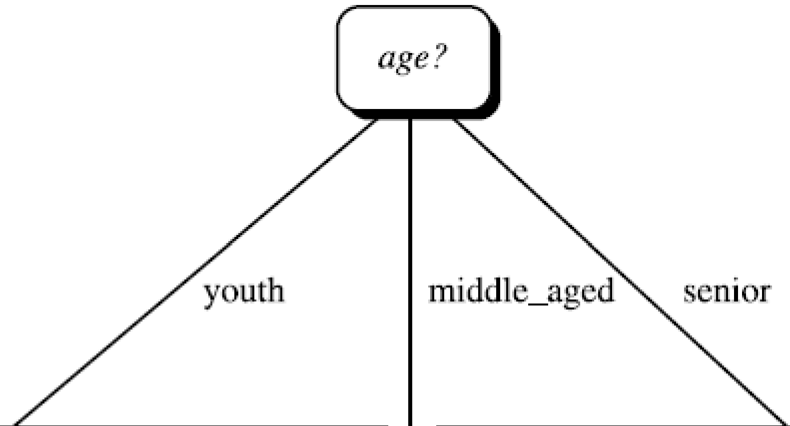
$$Info(D) = -\frac{9}{14}\log_2\left(\frac{9}{14}\right) - \frac{5}{14}\log_2\left(\frac{5}{14}\right) = 0.940 \text{ bits.}$$

$$\begin{aligned} Info_{age}(D) = & \frac{5}{14} \times \left(-\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5}\right) \\ & + \frac{4}{14} \times \left(-\frac{4}{4}\log_2\frac{4}{4} - \frac{0}{4}\log_2\frac{0}{4}\right) \\ & + \frac{5}{14} \times \left(-\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5}\right) \\ = & \ 0.694 \text{ bits.} \end{aligned}$$

$$Gain(age) = Info(D) - Info_{age}(D) = 0.940 - 0.694 = 0.246 \text{ bits.}$$

Gain(income)=0.029, Gain(student)=0.151,
Gain(credit_rating)=0.048

➔Age



youth:

income	*student*	*credit_rating*	*class*
high	no	fair	no
high	no	excellent	no
medium	no	fair	no
low	yes	fair	yes
medium	yes	excellent	yes

middle_aged:

income	*student*	*credit_rating*	*class*
high	no	fair	yes
low	yes	excellent	yes
medium	no	excellent	yes
high	yes	fair	yes

senior:

income	*student*	*credit_rating*	*class*
medium	no	fair	yes
low	yes	fair	yes
low	yes	excellent	no
medium	yes	fair	yes
medium	no	excellent	no

Tree Pruning

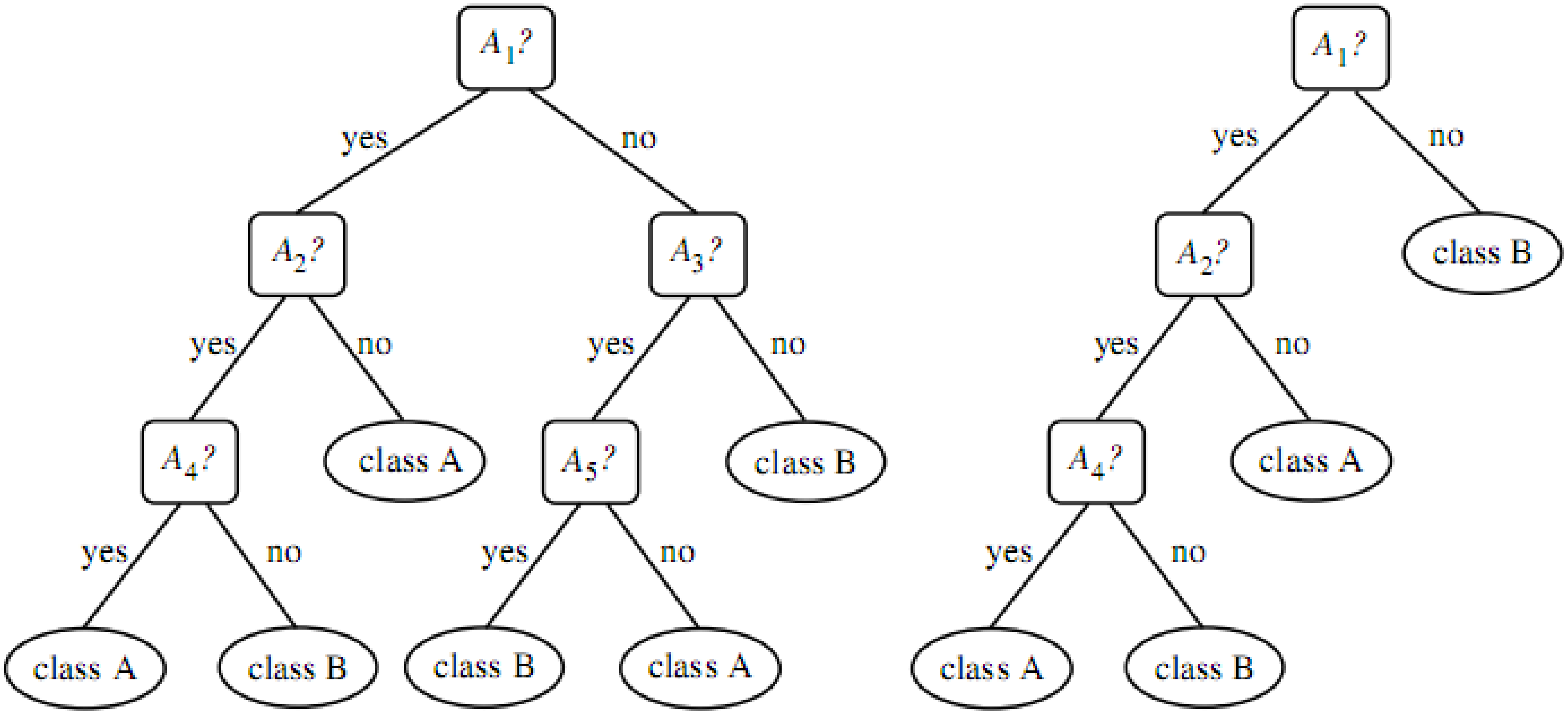
- When a decision tree is built, many of the branches will reflect anomalies in the training data due to noise or outliers. Tree pruning methods address this problem of *overfitting* the data.
- There are two common approaches for tree pruning
  - Pre-pruning (early stopping rule):
    - The tree is "pruned" by halting its construction early
    - Or stop algorithm before it becomes a fully grown tree
    - Most popular test: chi-squared test

Tree Pruning

– Post-pruning

- removes sub-trees from a "fully grown" tree: get a sequence of progressively pruned trees
- Possible strategies: error estimation, significance testing, MDL pruning
- preferred in practice

Example..



An unpruned decision tree and a pruned version of it.

Rule Based Classification

- has learned model as a set of IF-THEN rules
- We need to
  - How to generate the model
  - Examine how to use model to classify data
- IF-THEN rule is an expression of the form

  IF condition THEN conclusion

Ex:

R1:IF age=youth and student=yes THEN buys_computer=yes

Or

R1: (age=youth)∧(student=yes)=>(buys_computer=yes)

Using IF-THEN rules for classification (1/7)

- IF part is called rule antecedent or precondition, it can consist of one or more attributes test
- THEN part is called rule consequent, it consist a class prediction
- A rule R can be assessed by its coverage and accuracy
  - Given a tuple X from a data D
  - Let n_{cover}: # of tuples covered by R
  - $n_{correct}$: # of tuples correctly classify by R
  - $|D|$: # of tuples in D

➔

$$coverage(R) = \frac{n_{covers}}{|D|}$$

$$accuracy(R) = \frac{n_{correct}}{n_{covers}}.$$

Using IF-THEN rules for classification (2/7)

- Ex: of assessing R

Table 6.1 Class-labeled training tuples from the *AllElectronics* customer database.

RID	*age*	*income*	*student*	*credit_rating*	*Class: buys_computer*
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
3	middle_aged	high	no	fair	yes
4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	excellent	no
7	middle_aged	low	yes	excellent	yes
8	youth	medium	no	fair	no
9	youth	low	yes	fair	yes
10	senior	medium	yes	fair	yes
11	youth	medium	yes	excellent	yes
12	middle_aged	medium	no	excellent	yes
13	middle_aged	high	yes	fair	yes
14	senior	medium	no	excellent	no

Using IF-THEN rules for classification (3/7)

R: IF age=youth AND student=yes THEN buys_computer=yes

➔ $|D| = 14$

$n_{cover} = 2$

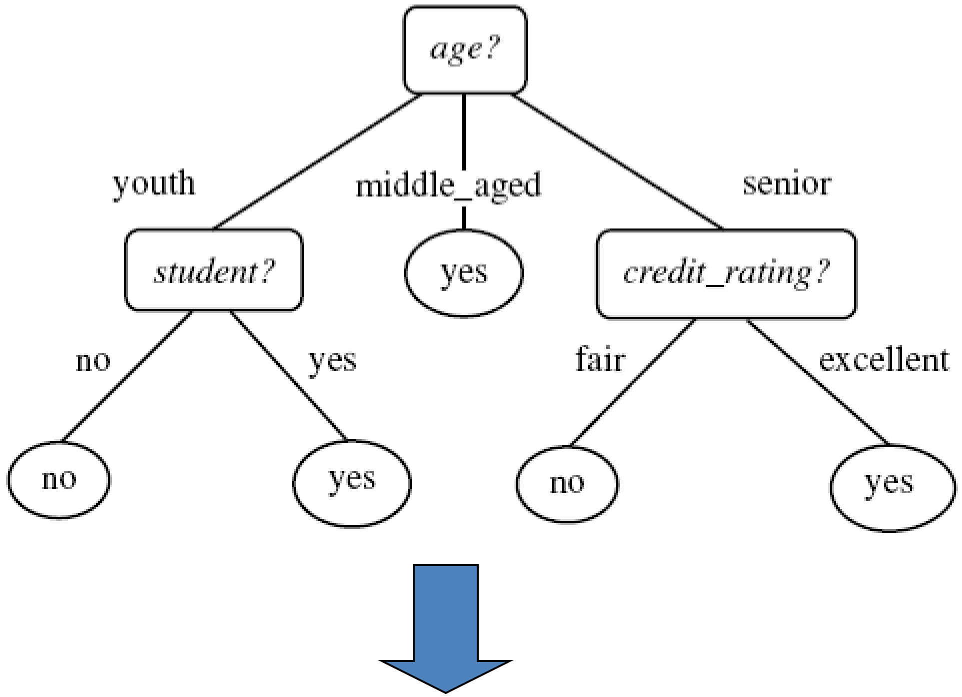
$n_{correct} = 2$

$$coverage(R) = \frac{2}{14} = 14.28\%$$

$$accuracy(R) = \frac{2}{2} = 100\%$$

Rule Extraction from a Decision Tree

- One rule is created for each path from the root to a leaf node
- Each splitting criterion is logically AND to form the rule antecedent (IF part)
- Leaf node holds the class prediction for rule consequent (THEN part)
- Logical OR is implied between each of the extracted rules



$R1$: IF $age = youth$ AND $student = no$ THEN $buys\_computer = no$

$R2$: IF $age = youth$ AND $student = yes$ THEN $buys\_computer = yes$

$R3$: IF $age = middle\_aged$ THEN $buys\_computer = yes$

$R4$: IF $age = senior$ AND $credit\_rating = excellent$ THEN $buys\_computer = yes$

$R5$: IF $age = senior$ AND $credit\_rating = fair$ THEN $buys\_computer = no$

Bayesian classification

- are statistical classifiers
- based on Baye's theorem
- Simple Bayesian classifier called naïve Bayesian classifier
- the effect of an attribute value on a given class is independent of the values of the other attributes: this assumption is called class conditional independence

Baye's theorem

- Is name after Thomas Bayes, nonconformist English clergyman who did early work in probability and decision theory during the 18th century

  $\in$

- Let X : a data tuple, evidence, measure on n attributes.

  H: some hypothesis such as that $X \in$ class C

$\Rightarrow$We want to calculate

P(H|X): prob that hypothesis H holds given evidence X or prob that $X \in C$

Baye's theorem

- **P(H|X)**: is a posterior prob or posteriori prob condition on X

Ex: X=(age=35, income=$40,000)

H: hypothesis that our customer will buy computer

=> P(H|X): prob that customer X will buy computer giving that we know their age and income

Baye's theorem

- Baye's theorem

$$P(H \mid X) = \frac{P(X \mid H)P(H)}{P(X)}$$

Naïve Bayesian Classification

- Works as follow

1. From data set D
   - Associated class label
   - n dimensional att vector X=(x1,x2,x3,…,xn), depiction n measurement made on the tuple from n atts. A1, A2, A3,…, An
2. Suppose we have m classes c1, c2, …,cm

Giving tuple X, classifier will predict X belong to highest

posterior probability, condition on X.

X $\in$Ci iif P(Ci|X) > P (Cj|X) for 1 <= j <=m, j I $\neq$

Ci, for which P(Ci|X) is maximized is called maximum

posterior hypothesis; $$P(C_i \mid X) = \frac{P(X \mid C_i)P(Ci)}{P(X)}$$

Naïve Bayesian Classification

3. P(X) is constant for all classes

$\Rightarrow$ Maximize $P(X|C_i)P(C_i)$

If class prior prob are not know, commonly assumed that $P(C_1)=P(C_2)=\ldots=P(C_m)$

=>maximize $P(X|C_i)$

Else maximize $P(X|Ci)P(Ci)$

$$P(Ci) = \frac{|C_{i,D}|}{|D|}$$

Naïve Bayesian Classification

4. Calculate P(X|Ci) is extremely expensive

Naïve assumes class conditional independence is made.

=>

$$P(X \mid Ci) = \prod_{k=1}^{n} (x_k \mid Ci)$$

$$= P(x_1 \mid Ci).P(x_2 \mid Ci)...P(x_n \mid Ci)$$

Where x_k is the value of att. A_k for X

If A is category

=> $$P(x_k \mid C_i) = \frac{\#of\_tuple\_of\_class\_Ci\_inD\_have\_value\_Xk}{|C_{i,D}|}$$

Example

Table 6.1 Class-labeled training tuples from the *AllElectronics* customer database.

RID	*age*	*income*	*student*	*credit_rating*	*Class: buys_computer*
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
3	middle_aged	high	no	fair	yes
4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	excellent	no
7	middle_aged	low	yes	excellent	yes
8	youth	medium	no	fair	no
9	youth	low	yes	fair	yes
10	senior	medium	yes	fair	yes
11	youth	medium	yes	excellent	yes
12	middle_aged	medium	no	excellent	yes
13	middle_aged	high	yes	fair	yes
14	senior	medium	no	excellent	no

- A={age, income, student, credit_rating}
- Class label buy_computer = Yes|No
- C1: buy_computer = Yes
- C2: buy_computer = No
- X=(age=youth, income=medium, student=y, credit-rating=fair)
- We need to maximize P(X|Ci)P(Ci)

$$P(buys\_computer = yes) = 9/14 = 0.643$$
$$P(buys\_computer = no) = 5/14 = 0.357$$

To compute $PX|C_i)$, for $i = 1, 2$, we compute the following conditional probabilities:

$$P(age = youth \mid buys\_computer = yes) = 2/9 = 0.222$$
$$P(age = youth \mid buys\_computer = no) = 3/5 = 0.600$$
$$P(income = medium \mid buys\_computer = yes) = 4/9 = 0.444$$
$$P(income = medium \mid buys\_computer = no) = 2/5 = 0.400$$
$$P(student = yes \mid buys\_computer = yes) = 6/9 = 0.667$$
$$P(student = yes \mid buys\_computer = no) = 1/5 = 0.200$$
$$P(credit\_rating = fair \mid buys\_computer = yes) = 6/9 = 0.667$$
$$P(credit\_rating = fair \mid buys\_computer = no) = 2/5 = 0.400$$

Using the above probabilities, we obtain

$$P(X|buys\_computer = yes) = P(age = youth \mid buys\_computer = yes) \times$$
$$P(income = medium \mid buys\_computer = yes) \times$$
$$P(student = yes \mid buys\_computer = yes) \times$$
$$P(credit\_rating = fair \mid buys\_computer = yes)$$
$$= 0.222 \times 0.444 \times 0.667 \times 0.667 = 0.044.$$

Similarly,

$$P(X|buys\_computer = no) = 0.600 \times 0.400 \times 0.200 \times 0.400 = 0.019.$$

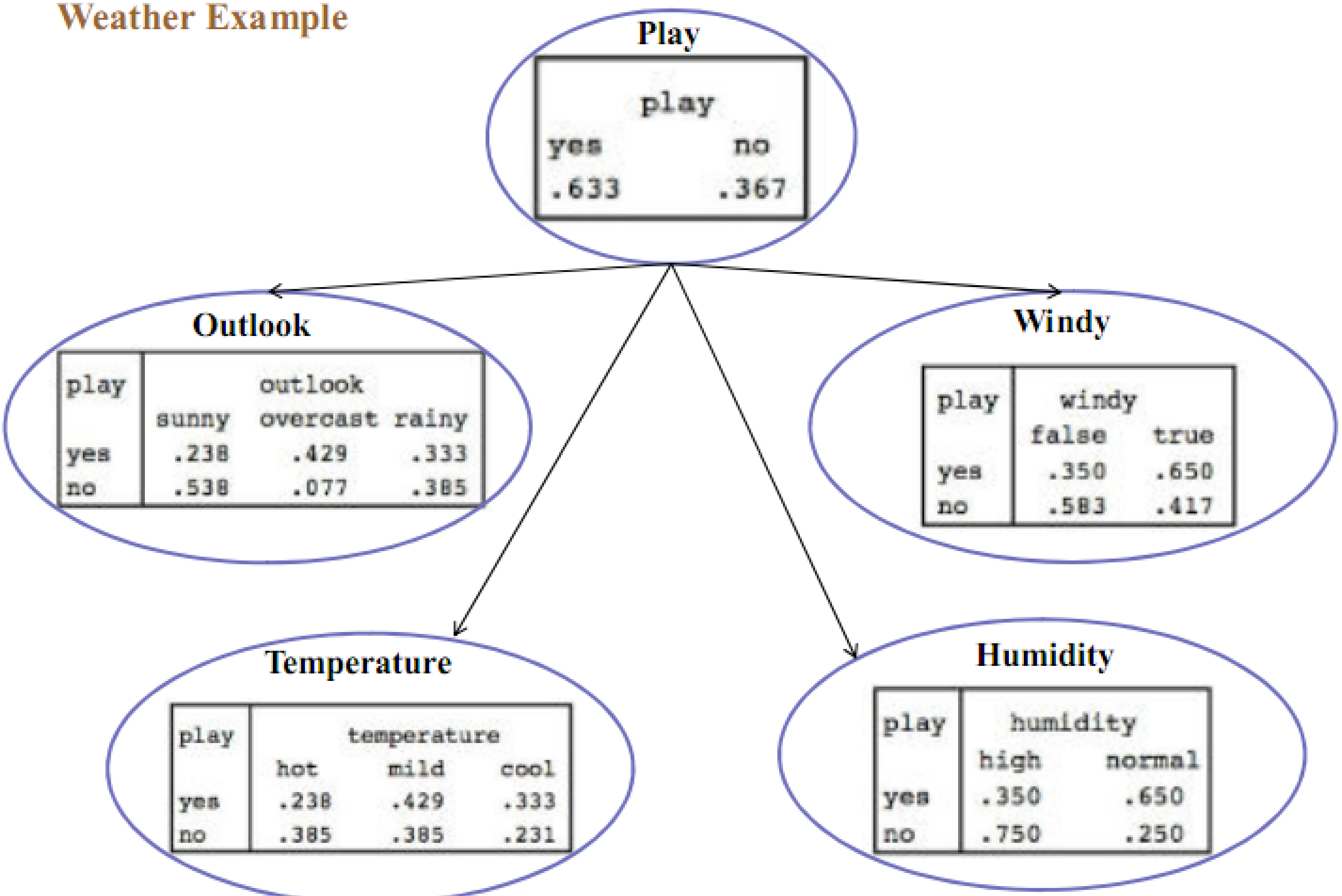
To find the class, C_i, that maximizes $P(X|C_i)P(C_i)$, we compute

$$P(X|buys\_computer = yes)P(buys\_computer = yes) = 0.044 \times 0.643 = 0.028$$
$$P(X|buys\_computer = no)P(buys\_computer = no) = 0.019 \times 0.357 = 0.007$$

Therefore, the naïve Bayesian classifier predicts $buys\_computer = yes$ for tuple X.

Weather Example

Weather Example



Play

play	
yes	no
.633	.367

Outlook

play	outlook		
	sunny	overcast	rainy
yes	.238	.429	.333
no	.538	.077	.385

Windy

play	windy	
	false	true
yes	.350	.650
no	.583	.417

Temperature

play	temperature		
	hot	mild	cool
yes	.238	.429	.333
no	.385	.385	.231

Humidity

play	humidity	
	high	normal
yes	.350	.650
no	.750	.250

Example

Suppose given X={outlook=rainy, temperature=cool, humidity=high, and windy=true), need to find out which play value (yes or no) should be assigned to X

Compare $p(\text{play=no}|X)$ and $p(\text{play=yes}|X)$

$p(\text{play=no}|X) = p(\text{play=no and } X) / p(X)$, $p(\text{play=yes}|X) = p(\text{play=yes and } X) / p(X)$

$p(\text{play=no and } X) = p(\text{play=no}) \times p(\text{outlook=rainy}|\text{play=no}) \times p(\text{temperature=cool}|\text{play=no}) \times p(\text{humidity=high}|\text{play=no}) \times p(\text{windy=true}|\text{play=no})$
$= 0.367 \times 0.385 \times 0.231 \times 0.750 \times 0.417 = 0.010$

$p(\text{play=yes and } X) = p(\text{play=yes}) \times p(\text{outlook=rainy}|\text{play=yes}) \times p(\text{temperature=cool}|\text{play=yes}) \times p(\text{humidity=high}|\text{play=yes}) \times p(\text{windy=true}|\text{play=yes})$
$= 0.633 \times 0.333 \times 0.333 \times 0.350 \times 0.650 = 0.016$

$p(\text{play=yes and } X) > p(\text{play=no and } X)$, so assign X to play value of "yes"

Advantages and Disadvantages Naïve Bayesian Classification

Advantages:

- Easy to implement
- Obtain good results in most of the cases

Disadvantages

- Assumption of class conditional independence usually doesn't hold
- Dependencies among these cannot be modeled by this classifier

How to deal with these Dependencies?

- Bayesian Belief Networks

Bayesian networks

- A simple, graphical notation for conditional independence assertions and hence for compact specification of full joint distributions

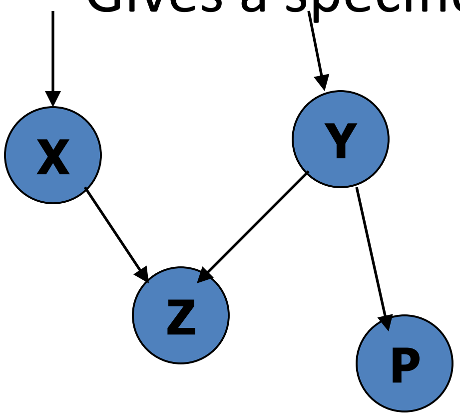
- Syntax:
  - a set of nodes, one per variable
  -
  - a directed, acyclic graph (link ≈ "directly influences")
  - a conditional distribution for each node given its parents:

$$\mathbf{P}(X_i \mid Parents(X_i))$$

- In the simplest case, conditional distribution represented as a **conditional probability table** (CPT) giving the distribution over X_i for each combination of parent values

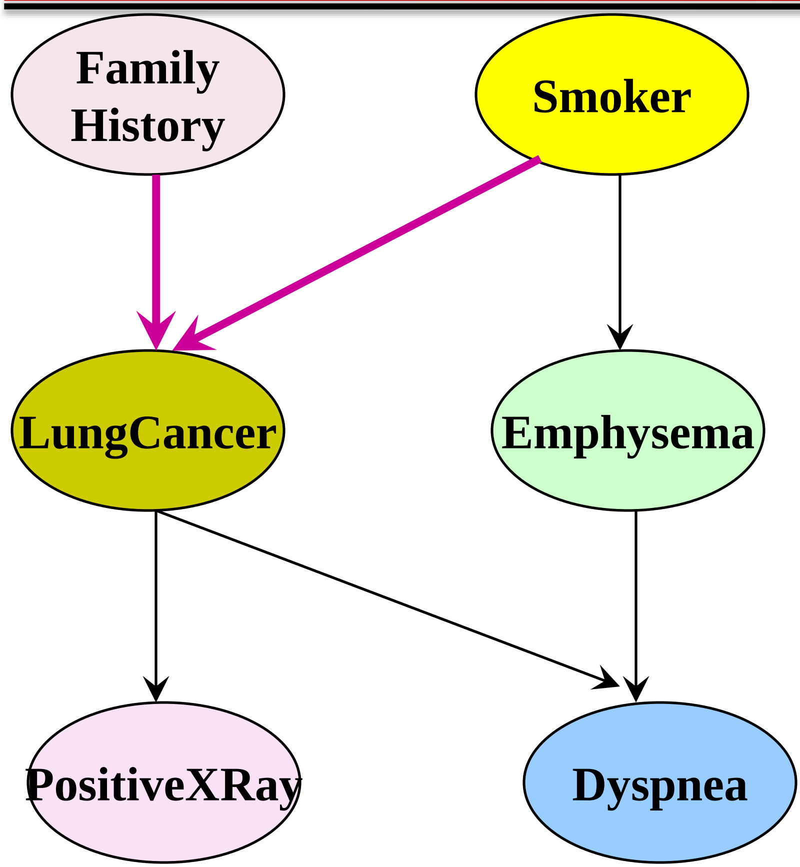
Bayesian Networks

- Bayesian belief network allows a *subset* of the variables conditionally independent
- A graphical model of causal relationships
  - Represents <u>dependency</u> among the variables
  - Gives a specification of joint probability distribution



❑Nodes: random variables
❑Links: dependency
❑X,Y are the parents of Z, and Y is the parent of P
❑No dependency between Z and P
❑Has no loops or cycles

Bayesian Belief Network: An Example



Bayesian Belief Networks

	(FH, S)	(FH, ~S)	(~FH, S)	(~FH, ~S)
LC	0.8	0.5	0.7	0.1
~LC	0.2	0.5	0.3	0.9

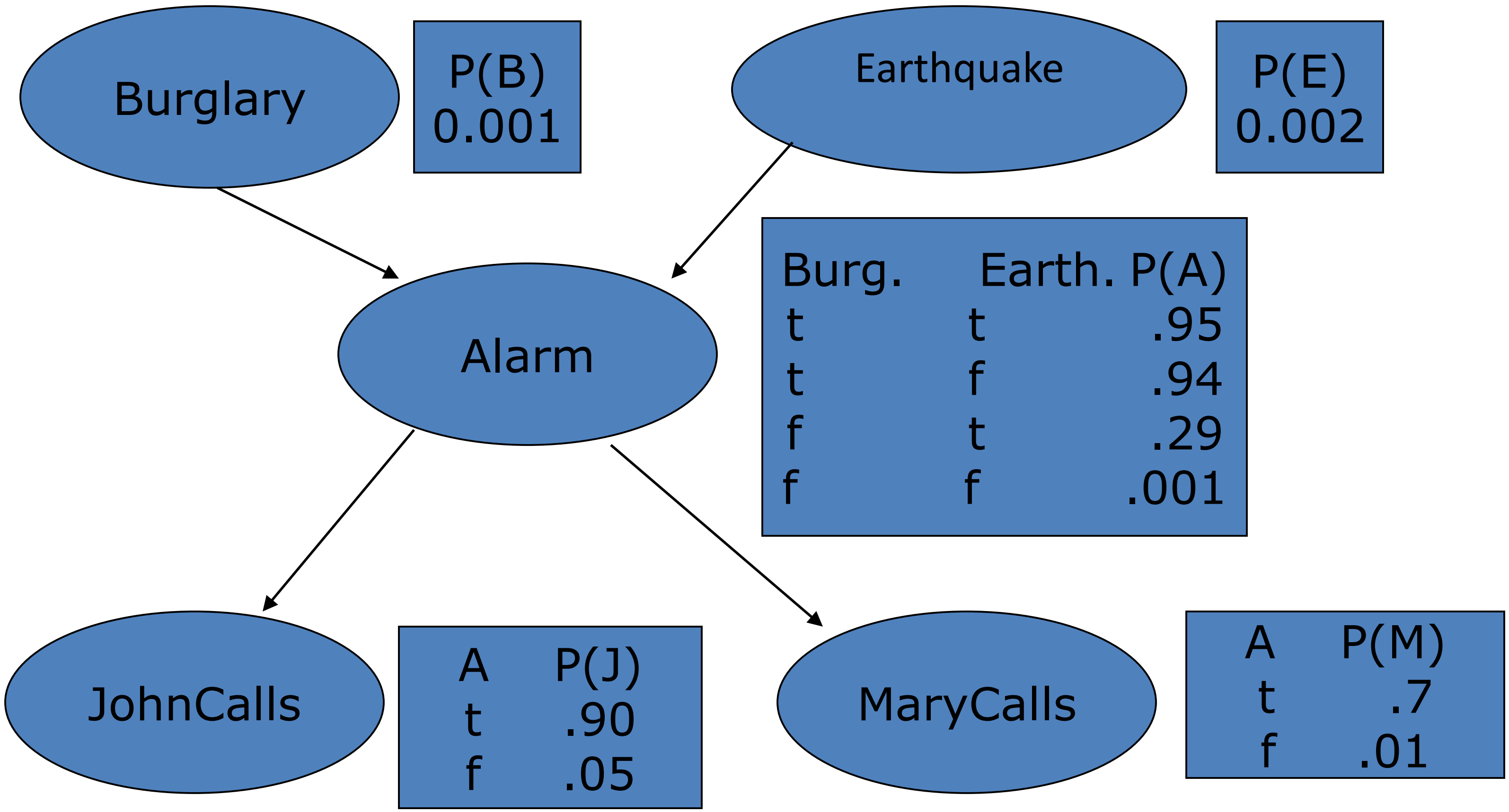
The **conditional probability table (CPT)** for the variable LungCancer:
Shows the conditional probability for each possible combination of its parents

$$P(z1,\ldots,zn) = \prod_{i=1}^{n} P(z_i \mid Parents(Z_i))$$

Example

- I'm at work, neighbor John calls to say my alarm is ringing, but neighbor Mary doesn't call. Sometimes it's set off by minor earthquakes. Is there a burglar?
- Variables: *Burglary*, *Earthquake*, *Alarm*, *JohnCalls*, *MaryCalls*
- Network topology reflects "causal" knowledge:
  - A burglar can set the alarm off
  - An earthquake can set the alarm off
  - The alarm can cause Mary to call
  - The alarm can cause John to call

Belief Networks



Case Based Reasoning

Faced this situation before?

- Oops the car stopped.
  - What could have gone wrong?
- Aah.. Last time it happened, there was no petrol.
  - Is there petrol?
    - Yes.
  - Oh but wait I remember the tyre was punctured
- This is the normal thought process of a human when faced with a problem which is similar to a problem he/she had faced before.

How do we solve problems?

- By knowing the steps to apply
  - from symptoms to a plausible diagnosis
- How does an expert solve problems?
  - uses same “book learning” as a novice
  - but quickly selects the right knowledge to apply
- Heuristic knowledge (“rules of thumb”)
  - *“I don’t know why this works but it does and so I’ll use it again!”*

So what?

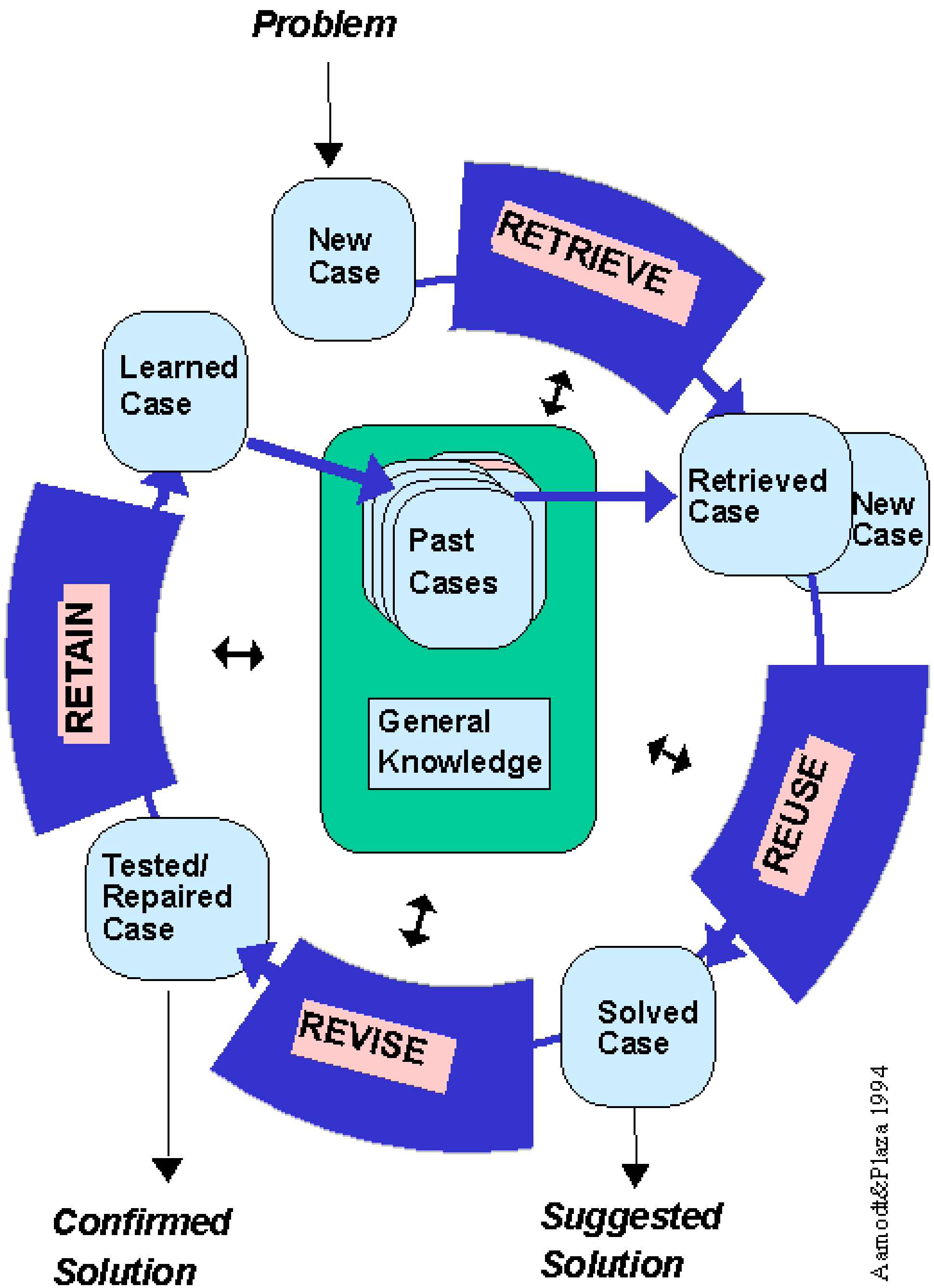
- Reuse the solution experience when faced with a similar problem.
- This is Case Based Reasoning (CBR)!
  - memory-based problem-solving
  - re-using past experiences
- Experts often find it easier to relate stories about past cases than to formulate rules

What’s CBR?

- To solve a new problem by remembering a previous similar situation and by reusing information and knowledge of that situation
- Ex: Medicine
  - doctor remembers previous patients especially for rare combinations of symptoms
- Ex: Law
  - case histories are consulted
- Ex: Management
  - decisions are often based on past rulings
- Ex: Financial
  - performance is predicted by past results

Definitions of CBR

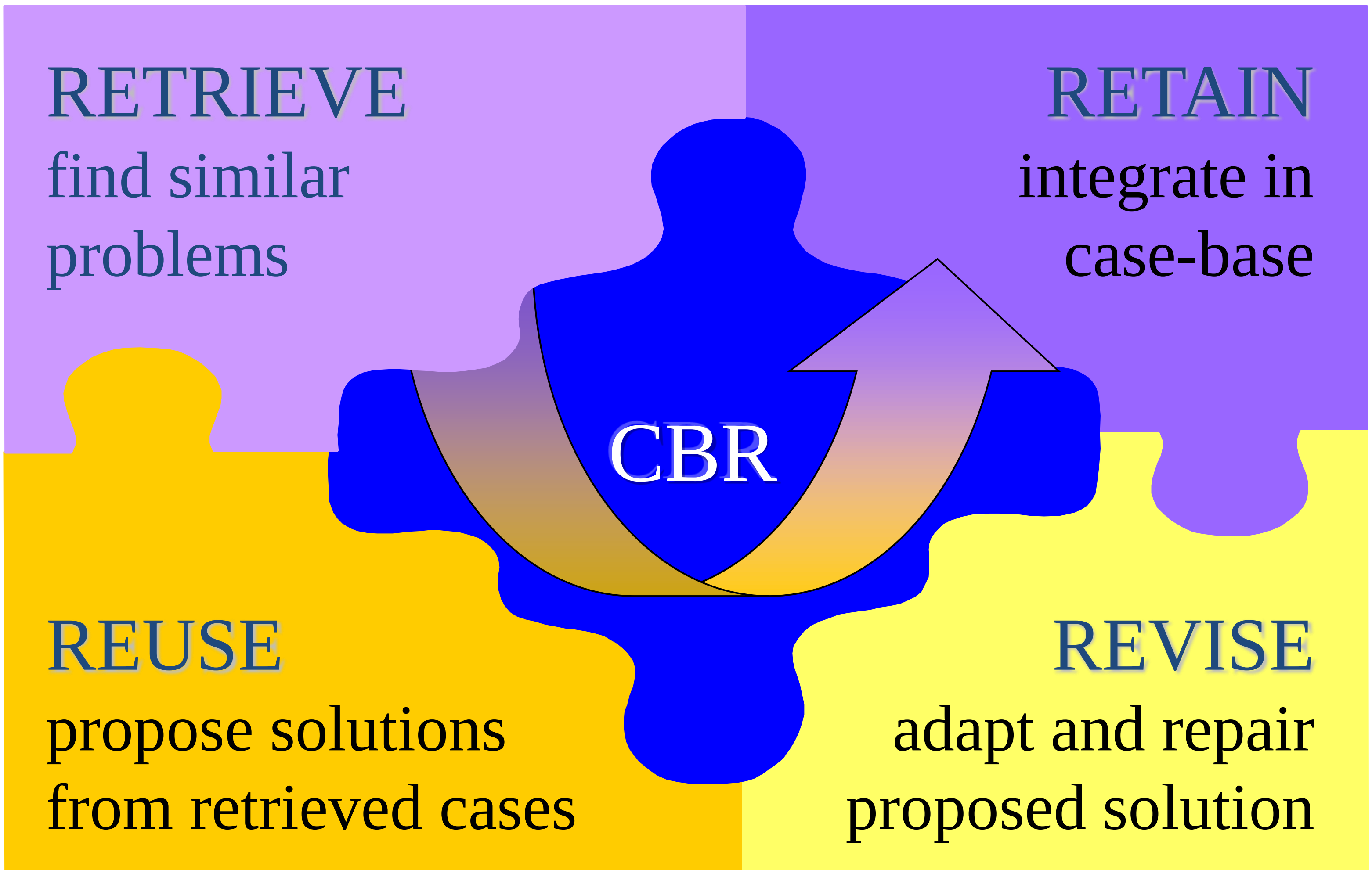
- Case-based reasoning is [...] reasoning by remembering - *Leake, 1996*
- A case-based reasoner solves new problems by adapting solutions that were used to solve old problems - *Riesbeck & Schank, 1989*
- Case-based reasoning is a recent approach to problem solving and learning [...] - *Aamodt & Plaza, 1994*



Problem
New Case
RETRIEVE
Learned Case
Past Cases
General Knowledge
Retrieved Case
New Case
RETAIN
REUSE
Tested/ Repaired Case
REVISE
Solved Case
Confirmed Solution
Suggested Solution
Aamodt&Plaza 1994

CBR Cycle

R^4 Cycle


RETRIEVE
find similar problems
RETAIN
integrate in case-base
CBR
REUSE
propose solutions from retrieved cases
REVISE
adapt and repair proposed solution

CBR System Components

- Case-base
  - database of previous cases (experience)
- Retrieval of relevant cases
  - index for cases in library
  - matching most similar case(s)
  - retrieving the solution(s) from these case(s)
- Adaptation of solution
  - alter the retrieved solution(s) to reflect differences between new case and retrieved case(s)

CBR Assumption(s)

- The main assumption is that:
  - *Similar problems have similar solutions:*
    - e.g., an aspirin can be taken for any mild pain
- Two other assumptions:
  - The world is a regular place: what holds true today will probably hold true tomorrow
    - (e.g., if you have a headache, you take aspirin, because it has always helped)
  - Situations repeat: if they do not, there is no point in remembering them
    - (e.g., it helps to remember how you found a parking space near that restaurant)

Technical Diagnosis of Car Faults

Feature Value

CASE 1

Problem (Symptoms)

- *Problem*: Front light doesn't work
- *Car*: VW Golf II, 1.6 L
- *Year*: 1993
- *Battery voltage*: 13,6 V
- *State of lights*: OK
- *State of light switch*: OK

Solution

- *Diagnosis*: Front light fuse defect
- *Repair*: Replace front light fuse

CASE 2

Problem (Symptoms)

- Problem: Front light doesn't work
- Car: Audi A6
- Year: 1995
- Battery voltage : 12,9 V
- State of lights: surface damaged
- State of light switch: OK

Solution

- Diagnosis: Bulb defect
- Repair: Replace front light

What Is Prediction?

- (Numerical) prediction is similar to classification
  - construct a model
  - use model to predict continuous or ordered value for a given input
- Prediction is different from classification
  - Classification refers to predict categorical class label
  - Prediction models continuous-valued functions
- Major method for prediction: regression
  - model the relationship between one or more *independent* or **predictor** variables and a *dependent* or **response** variable
- Regression analysis
  - Linear and multiple regression
  - Non-linear regression
  - Other regression methods: generalized linear model, Poisson regression, log-linear models, regression trees

Linear Regression

- Linear regression: involves a response variable y and a single predictor variable x

  $y = w_0 + w_1 x$

  where w_0 (y-intercept) and w_1 (slope) are regression coefficients
- Method of least squares: estimates the best-fitting straight line

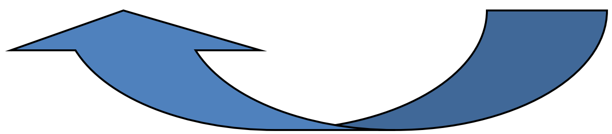
$$w_1 = \frac{\sum_{i=1}^{|D|}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{|D|}(x_i - \bar{x})^2} \qquad w_0 = \bar{y} - w_1 \bar{x}$$

- Multiple linear regression: involves more than one predictor variable
  - Training data is of the form $(\mathbf{X_1}, y_1), (\mathbf{X_2}, y_2), \ldots, (\mathbf{X_{|D|}}, y_{|D|})$
  - Ex. For 2-D data, we may have: $y = w_0 + w_1 x_1 + w_2 x_2$
  - Solvable by extension of least square method or using SAS, S-Plus
  - Many nonlinear functions can be transformed into the above

Linear Regression

A regression model is comprised of a dependent, or response, variable and an independent, or predictor, variable.

Dependent Variable = Independent Variable(s)

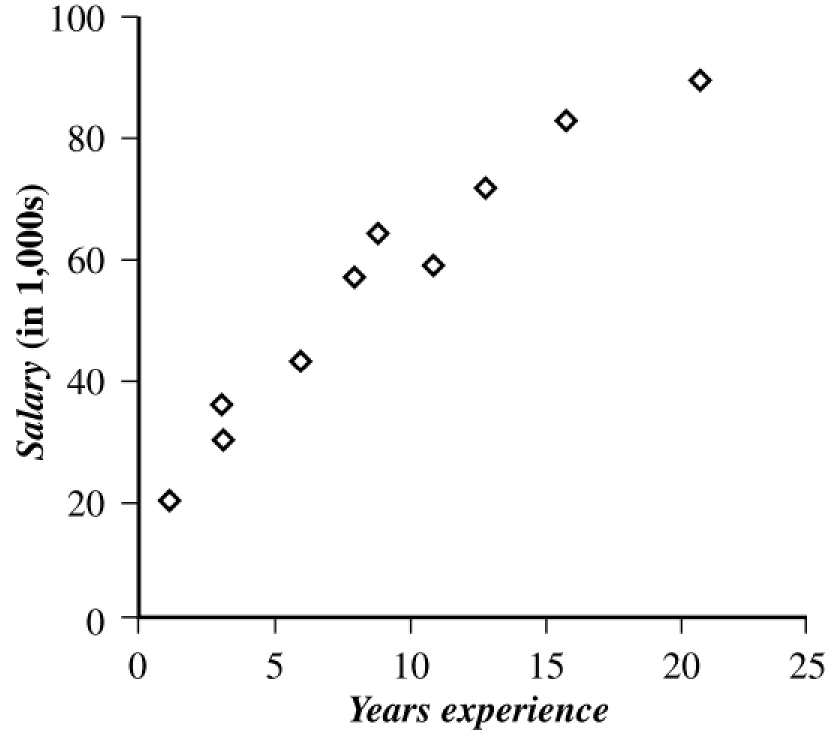


Prediction Relationship

Linear Regression

Salary data.

x years experience	*y salary* (*in* $*1000s*)
3	30
8	57
9	64
13	72
3	36
6	43
11	59
21	90
1	20
16	83



Plot of data: Although the points do not fall on a straight line, the overall pattern suggests a linear relationship between x(years experience) and y (salary)

Linear Regression

Given the above data, we compute $\bar{x} = 9.1$ and $\bar{y} = 55.4$. Substituting these values into Equations (6.50) and (6.51), we get

$$w_1 = \frac{(3-9.1)(30-55.4)+(8-9.1)(57-55.4)+\cdots+(16-9.1)(83-55.4)}{(3-9.1)^2+(8-9.1)^2+\cdots+(16-9.1)^2} = 3.5$$

$$w_0 = 55.4-(3.5)(9.1) = 23.6$$

Thus, the equation of the least squares line is estimated by $y = 23.6+3.5x$. Using this equation, we can predict that the salary of a college graduate with, say, 10 years of experience is \$58,600. ■