ECpE 4524

Knowledge-Based Systems

often called

Expert Systems

Knowledge-based systems

(textbook, chapter 20)

Goal:

Try to solve the kinds of problems that normally require human experts

Typical examples:

medical diagnosis, financial analysis, factory production scheduling

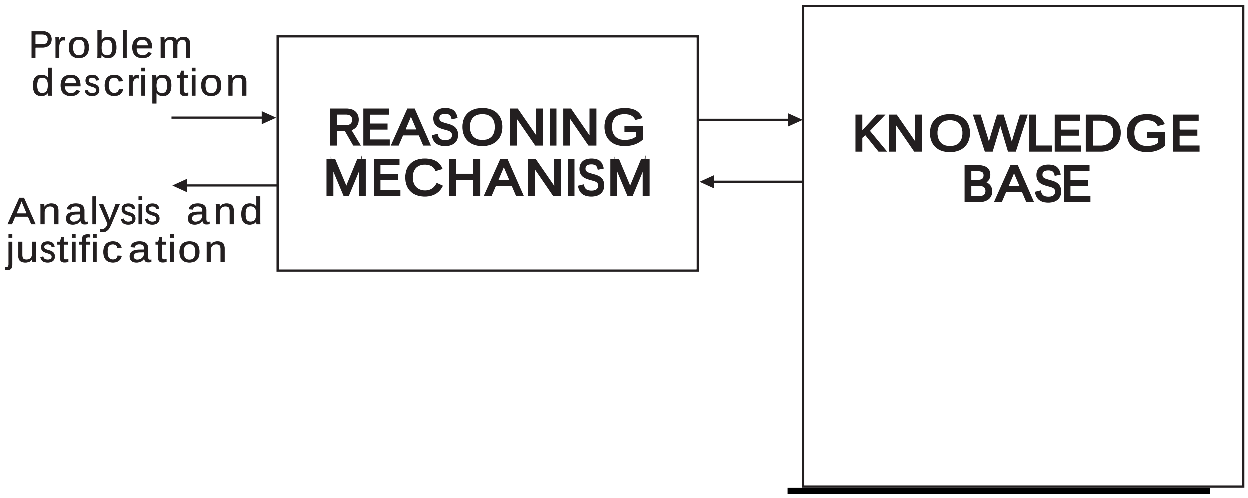
Why study knowledge-based systems?

To understand human reasoning methods

Human experts tend to take vacations, get hired by other companies, ask for raises, retire, become ill, die, . . .

Lots of commercial successes!

Expert system overview:



The **knowledge base** . . .

- contains "domain knowledge," normally provided by human experts
- is typically very specialized for a particular problem domain
- is often encoded as IF-THEN rules
- may incorporate heuristics or probabilities
- is a <u>valuable</u> commodity

Building, validating, and maintaining a knowledge base is a skill (art) called ***knowledge engineering***

The **reasoning mechanism** . . .

- takes descriptions from the user about the problem to be solved
- requests additional information from the user as needed
- interprets the knowledge base to make inferences, draw conclusions, and ultimately give advice
- explains its reasoning to the user (how were the conclusions reached?)
- is sometimes called an **inference engine**

An example:

PUFF *(1979)*

Pulmonary function analysis

Physician refers patient to pulmonary testing lab

Patient inhales/exhales through tube connected to computerized instrument which measures flow rates and air volumes

PUFF accepts this data along with auxiliary data (age, sex, smoking history), and prints diagnosis in English

Now used on a routine basis (?)

Building PUFF's knowledge base:
A knowledge engineer sat down with an expert pulmonary physiologist at the Pacific Medical Center in San Francisco and developed rules (64 in all)

IF 1. ----------
2. ----------
3. ----------
. . .

THEN 1. ----------
2. ----------
. . .

(A more recent version had about 400 rules.)

Example PUFF rule:

RULE31

IF:

1. The severity of obstructive airways disease of the patient is greater than or equal to mild, and
2. the degree of diffusion defect of the patient is greater than or equal to mild, and
3. the TLC observed/predicted of the patient is greater than or equal to 110, and
4. the observed/predicted difference in RV/TLC of the patient is greater than or equal to 10

THEN:

1. There is strongly suggestive evidence (0.9) that the subtype of obstructive airways disease is emphysema, and
2. It is definite (1.0) that "OAD, Diffusion Defect, elevated TLC, and elevated RV together indicate emphysema" is one of the findings

Sample PUFF session:

PATIENT DATA:
The degree of dyspnea: MODERATELY-SEVERE
The severity of coughing: MILD
Sputum production: MODERATELY-SEVERE
The number of pack-years of smoking: 48
Referal diagnosis: BRONCHITIS
IVC/IVC-predicted: 80
RV/RV-predicted: 191
FVC/FVC-predicted: 87
TLC observed/predicted: 127
Predicted FEV1/FVC: 83
TLC(DLCO) observed/predicted: 83
FEV1/FVC ratio: 50
RV/TLC observed/predicted: 21
MMF/MMF-predicted: 19
DLCO/DLCO-predicted: 48
The slope of (F50obs-F25obs)/FVCobs: 19

DEGREE OF OBSTRUCTIVE AIRWAYS DISEASE:
OAD degree by slope: MODERATELY-SEVERE 700
OAD degree by MMF: SEVERE 900
OAD degree by FEV1: MODERATELY-SEVERE 700
Final OAD degree: MODERATELY-SEVERE 910
    SEVERE 900

INTERPRETATION:

Obstruction is indicated by curvature of the flow-volume loop.

Forced Vital Capacity is normal and peak flow rates are reduced, indicating severe airway obstruction.

Change in expired flow rates following bronchodilation shows that there is reversibility of airway obstruction.

Elevated lung volumes indicate overinflation.

Air trapping is indicated by the elevated difference between observed and predicted RV/TLC ratios.

Airway obstruction is consistent with the patient's smoking history.

The airway obstruction accounts for the patient's dyspnea.

Although bronchodilators were not useful in this one case, prolonged use may prove to be beneficial.

Obstructive Airways Disease of mixed types.

How were the rules produced?

100 cases (previously diagnosed patients) were selected

The cases were chosen to span the variety of known disease states

The pulmonary function expert posed hypothetical rules for diagnosing the illness

The knowledge engineer encoded the rules (in LISP) and tested them with the test cases.

The expert reviewed the test results and modified or added rules to handle the cases that were incorrectly diagnosed

Looping continued until the expert was satisfied

How to test PUFF's performance?

150 additional different cases were analyzed
1) by human experts and
2) by PUFF

The diagnoses were compared:
90% matched to same degree of severity
100% matched to within one degree of severity

Effort:
50 hours by the expert
400 hours by the knowledge engineer

The 64 rules were "popped into" an existing expert system

OBSERVATIONS

Human experts are often unaware of how they reach conclusions

The expert usually knows more than he/she is aware of knowing

The knowledge brought to bear by the expert is often experiential, heuristic, and uncertain

General problem-solvers (domain-independent) are too weak for building real-world, high-performance systems

The behavior of the best problem-solvers (humans) is weak and shallow except in areas of specialization

Expertise in one specialization area usually does not transfer well to other areas

Recall <u>weak</u> vs. <u>strong</u> methods:

Weak methods
domain-independent, general-purpose
(example: GPS)

Strong methods
domain-specific, knowledge-rich
(examples: knowledge-based systems)

Example expert systems

Medicine

MYCIN (1976)

Identification of bacteria in blood and urine samples; prescription of antibiotics

INTERNIST / CADUCEUS (1970s / 1984)

Diagnosis of majority of diseases in field of internal medicine

PUFF (1979)

Interpretation of respiratory tests for diagnosis of pulmonary disorders

BABY (19??)

Patient monitoring in a newborn intensive care unit

QMR (1988) (Quick Medical Record)

Assists physicians in diagnosis of over 4000 disease manifestations (uses the INTERNIST knowledge base)

CHEMISTRY

DENDRAL (1960s and 1970s)
Identification of molecular structure of organic compounds
CRYSALIS (19??)
Interpretation of electron density maps in protein crystallography
MOLGEN (19??)
Planning DNA-manipulation experiments in molecular genetics

AGRICULTURE

PLANT/ds
Diagnosing diseases in soybeans
PLANT/cd
Diagnosing cutworm damage in corn

OTHERS

PROSPECTOR (1978)

Provides advice on mineral prospecting

MACSYMA (1968 - present)

Symbolic solutions to mathematical problems

R1 / XCON (1982)

Configures VAX computer systems

GATES (1988)

Used by TWA at JFK airport to assist ground controllers in assigning gates to arriving and departing flights

DESIGN ADVISOR (1989)

Critiques IC designs

TOP SECRET (1989)

Decide the correct security classification to give a nuclear weapons document

DENDRAL

Feigenbaum (1960s and 70s)

One of the first expert systems

Identifies of molecular structure of organic compounds

Uses mass spectrogram and nuclear magnetic resonance (NMR) data

MYCIN (a precursor to PUFF)

(textbook, Section 8.2)

Shortliffe, 1976 (Stanford, in Interlisp)

MYCIN is possibly the best known expert system that has been developed

MYCIN can diagnose bacterial infections and recommend treatment

MYCIN was designed solely as a research effort (never used in practice?);
it served as a model for much subsequent expert-system work

MYCIN uses IF-THEN rules

Each rule has a **certainty factor** associated with it (a measure of the confidence associated with the rule's conclusion)

example

RULE50

IF 1) the infection is primary-bacteremia, and

2) the site of the culture is one of the sterile sites, and

3) the suspected portal of entry of the organism is the gastrointestinal tract

THEN

there is suggestive evidence (0.7) that the identity of the organism is bacteroides *in LISP:*

PREMISE:

```
(AND
    (SAME CNTXT INFECT
    PRIMARY-BACTEREMIA)
    (MEMBF CNTXT SITE
    STERILISITES)
    (SAME CNTXT PORTAL GI) )
```

ACTION:

```
(CONCLUDE CNTXT IDENT
    BACTEROIDES TALLY 0.7)
```

Each rule:

premise $\Rightarrow$ action

Each premise:

logical conjunction of clauses

Each clause:

<predicate> <object> <attribute> <value>

MYCIN used a front-end known as

TEIRESIAS

(textbook, section 20.3)

Davis, 1977, 1982

TEIRESIAS is a knowledge-acquisition manager; it provides explanations of how conclusions are reached

Example TEIRESIAS rule
(actually a meta-rule)

IF

1) the patient is a compromised host, and

2) there are rules that mention in their premise pseudomonas, and

3) there are rules that mention in their premise klebsiellas

THEN

there is suggestive evidence (0.4) that the former should be done before the latter

Example MYCIN / TEIRESIAS session:

1) Patient's name:
** Fred Smith
2) Sex
** male
3) Age:
** 55
4) Have you been able to obtain positive cultures from a site at which Fred Smith has an infection?
** yes
5) What is the infection?
** Primary-bacteremia
6) Please give the date and approximate time when signs or symptoms of the primary-bacteremia (infection-1) appeared.
** May 5, 1980
7) From what site was the specimen for culture-1 taken?
** Blood
8) Please give the date and time when this blood culture (culture-1) was obtained.
** May 9, 1980

9) Enter the identity of organism-1.
** unknown
10) Is organism-1 a rod, or coccus (etc.)?
** rod
11) The gram stain of organism-1:
** grammeg
12) Have there been positive cultures from sites which are NOT associated with infections about which you are seeking advice?
** no
13) Did organism-1 grow aerobically?
** yes (0.8)
14) Did organism-1 grow anerobically?
** yes
15) What is the suspected portal of entry of organism-1 into the sterile site?
** GI (0.6)

. . . etc.

My therapy recommendation will be based on the following probable infection(s) and potential causative organism(s):

INFECTION-1 is PRIMARY-BACTEREMIA
<Item 1> Organism-1 may be PSEUDOMONAS-AERUGINOS
<Item 2> Organism-1 may be KLEBSIELLA-PNEUMONIAE
<Item 3> Organism-1 may be E. COLI
<Item 4> Organism-1 may be BACTEROIDES-FRAGILIS
<Item 5> Organism-1 may be ENTEROBACTER
<Item 6> Organism-1 may be PROTEUS-NON-MIRABILIS

46) On a scale of 0 to 4, where higher numbers indicate increasing severity, how would you rate Fred Smith's degree of sickness?
** 3
47) Does Fred Smith have a clinically significant allergic reaction to any antimicrobial agent?
** no
48) Fred Smith's weight in kilograms:
** 70

[Rec 1] My preferred therapy recommendation is as follows:

In order to cover for Items <1 2 3 5 6>
Give: GENTAMICIN
Dose: 119 mg q8h IV [or IM] for 10 days
Comments: Modify dose in renal failure

In order to cover for Item <4>
Give: CLINDAMYCIN
Dose: 595 mg q6h IV [or IM] for 14 days
Comments: If diarrhea or other GI symptoms develop, patient should be evaluated for possible pseudomembranous colitis

Do you wish to see the next-choice therapy?
** yes
Which, if any, of the above drugs do you wish eliminated from consideration?
** Clindamycin

[Rec 2]
In order to cover for Items <2 3 4 5 6>
Give: CHLORAMPHENICOL
Dose: 1 mg q6h IV for 14 days
Comments: Monitor patient's white count

In order to cover for Item <1>
Give: GENTAMICIN
Dose: 119 mg q8h IV [or IM] for 10 days
Comments: Modify dose in renal failure.

The ability to explain the reasoning path is important

(MYCIN:) Is the patient's illness with organism-1 a hospital-acquired infection?

(USER:) ** why

(TEIRESIAS:) (WHY is it important to determine whether or not the infection with organism-1 was acquired while the patient was hospitalized?)

[1.0] This will aid in determining the category of organism-1. It has already been established that

[1.1] the gram stain of organism-1 is gramneg, and

[1.2] the morphology of organism-1 is rod, and

[1.3] the aerobicity of organism-1 is faculative

Therefore, if

[1.4] the infection with organism-1 was not acquired while the patient was hospitalized

then there is strongly suggestive evidence (0.8) that the category of organism-1 is enterobacteriaceae.

Also: there is weakly suggestive evidence (0.1) that the identify of organism-1 is pseudomonos-aeruginosa [RULE37].

(USER:) how 1.3

(TEIRESIAS:) (HOW was it established that the aerobicity of organism-1 is faculative?)

The following were used:

[3.1] RULE27 indicated there is strongly suggestive evidence (0.8) that the aerobicity of organism-1 is faculative.

[3.2] RULE27 indicated there is weakly suggestive evidence (0.2) that the aerobicity of organism-1 is anerobic

Since this gave a cumulative CF of (0.8) for faculative, and (0.2) for anerobic, it has been established that the aerobicity of organism-1 is faculative.

(USER:) why didn't you consider streptococcus as a possibility for organism-1?

(TEIRESIAS:) The following rule could have been used to determine that the identity of organism-1 was streptococcus: RULE33.

But clause 2 ("the morphology of the organism is coccus") was already known to be false for organism-1, so the rule was never tried.

How does MYCIN know what questions to ask?

At each point, the question is determined by MYCIN's current hypothesis (and answers to previous questions)

MYCIN is a backward-chaining system: Eg., to determine the cause of the patient's illness, MYCIN looks for rules which have a THEN clause suggesting diseases;
MYCIN then uses the IF clauses to set up subgoals, and looks for THEN clauses of other rules to satisfy these subgoals, etc.

This approach makes it easier for the physician to follow the "thought" process, and it simplifies the English-language interface

MYCIN summary

... recommends therapies for patients with bacterial infections

... uses IF-THEN rules (with certainty factors) to represent knowledge

... interacts with a physician to acquire clinical data

... asks questions based on current hypothesis and known data

... reasons backward from its goal of recommending a therapy for a particular patient

... stores approx. 500 IF-THEN rules, and can recognize about 100 causes of bacterial infection

TEIRESIAS summary

... serves as a front-end to MYCIN
... was the first program to provide explanations of how conclusions were reached
... intercepts questions such as "why" and "how" from the physician (i.e., <u>why</u> does MYCIN want certain information, and <u>how</u> did MYCIN reach a certain conclusion)
... TEIRESIAS can answer "why" questions by examining its internal tree of subgoals
... TEIRESIAS can answer "how" questions by identifying the pieces of evidence that supported MYCIN's IF clauses

Expert system shells

After MYCIN was built, someone observed that the knowledge base could be replaced by completely new rules

MYCIN without its knowledge base was called EMYCIN (Empty MYCIN) (and was used to implement PUFF)

Today you can buy similar "shells" that contain a user interface, a reasoning subsystem, and an explanation subsystem

With such a shell, the user can concentrate on the knowledge base

In many expert systems, the rules are written as follows:

$$\textbf{symptom} \Rightarrow \textbf{disease}$$

(the diagnosis must work from symptoms to find the cause)

But in reality, we know that

$$\textbf{disease} \Rightarrow \textbf{symptom}$$

Abductive reasoning is <u>not</u> truth-preserving:

$$\begin{array}{l} P \Rightarrow Q \\ Q \\ \hline \therefore P \end{array}$$

Reasoning under uncertainty

(Inexact reasoning)

We can attach "confidence" or "belief" values to

- the inference itself:
  $A \Rightarrow B$ (with confidence 0.8)

- the evidence:
  A (which has confidence 0.6)
  $\Rightarrow B$

- both

Our first impulse for inexact reasoning: use *probability theory!*

What is Pr(measles | spots) ?

Recall Bayes' theorem:

$$\Pr(measles|spots) = \frac{\Pr(spots|measles)\Pr(measles)}{\Pr(spots)}$$

Looks fine. Now we'd like to consider other possible diseases:

$$\Pr(H_i|spots) = \frac{\Pr(spots|H_i)\Pr(H_i)}{\Pr(spots)}$$

If the diseases are <u>exhaustive</u> and <u>mutually exclusive:</u>

$$= \frac{\Pr(spots|H_i)\Pr(H_i)}{\sum_i \Pr(spots|H_i)\Pr(H_i)}$$

Now consider two different symptoms for one disease:

$$\Pr(H_i | spots \wedge fever) = \frac{\Pr(spots \wedge fever | H_i)\Pr(H_i)}{\Pr(spots \wedge fever)}$$

Problem: how do we compute these ?

$$\Pr(spots \wedge fever)$$

$$\Pr(spots \wedge fever | H_i)$$

It is common (and absurd!) to assume that spots and fever are <u>independent</u>:

$$\Pr(spots \wedge fever) = \Pr(spots)\Pr(fever)$$

To really use Bayes' theorem, we would need probabilities for all possible combinations of symptoms in all conditional expressions: ***not feasible!***

Standard reasons why Bayesian reasoning <u>cannot work</u>:

- in "pure form" it requires an impossible number of probabilities
- the usual remedy is to impose absurd assumptions of independence
- knowing *any* probability may be unrealistic
  (usually just use statistical frequency)
- it only works for the single-disease situation

Still, it's a good starting point . . .

MYCIN's Confidence Factors

a MYCIN rule: $E \Rightarrow H$ (CF=x)

Confidence Factor:

1.0 true with complete confidence

-1.0 false with complete confidence

If x = 1.0 and E is a predicate, then we have normal logic

$$CF(H|E) = MB(H|E) - MD(H|E)$$

MB: "measure of belief"
MD: "measure of disbelief"

Each is in range [0, 1]

When one is nonzero, the other is normally zero

Consider $\mathrm{E}_1 \wedge \mathrm{E}_2 \Rightarrow \mathrm{H}$ (CF = *x*)

If the E_i are all certain, then H has CF = *x*

If the E_i are <u>not</u> all certain, then we need to "fold together" the confidence factors

For conjunctive evidence:

$$MB(E_1 \wedge E_2) = \min(MB(E_1), MB(E_2))$$

$$MD(E_1 \wedge E_2) = \max(MD(E_1), MD(E_2))$$

Now consider $\mathrm{E}_1 \vee \mathrm{E}_2 \Rightarrow \mathrm{H}$ (CF = *x*):

For disjunctive evidence:

$$MB(E_1 \vee E_2) = \max(MB(E_1), MB(E_2))$$

$$MD(E_1 \vee E_2) = \min(MD(H_1), MD(H_2))$$

What CF do we assign to H, for uncertain evidence P?

$\mathrm{P} \Rightarrow \mathrm{H}$ (CF = *x*)

$$MB(H) = MB'(H)\max(0, CF(P))$$

$$MD(H) = MD'(H)\max(0, CF(P))$$

Now consider this:

Rule 1: $\mathrm{E}_1 \Rightarrow \mathrm{H}$ (CF=x)

Rule 2: $\mathrm{E}_2 \Rightarrow \mathrm{H}$ (CF=y)

If both E_i are true,
then <u>both</u> should contribute to the confidence that H is true:

$$MB(H|E_1 \wedge E_2) = \begin{cases} 0 & MD(H|E_1 \wedge E_2) = 1 \\ MB(H|E_1) + MB(H|E_2) & otherwise \\ - MB(H|E_1)MB(H|E_2) & \end{cases}$$

$$MD(H|E_1 \wedge E_2) = \begin{cases} 0 & MB(H|E_1 \wedge E_2) = 1 \\ MD(H|E_1) + MD(H|E_2) & otherwise \\ - MD(H|E_1)MD(H|E_2) & \end{cases}$$

(see text, p. 234)

In MYCIN, rules are invoked by backwards-chaining using exhaustive depth-first search

Eg., find all rules that conclude the identity of an organism

Eg., see if all conditions are met; if not, set up subgoals (based on IF clauses)

If $-0.2 < CF < 0.2$, the CF value is regarded as unknown. In this case, MYCIN asks the user.

a different approach . . .

PLANT/ds

an expert system for diagnosing soybean diseases

Rule form: extended propositional logic

PLANT/ds rules

Let x1, x2, ..., xn represent different “features” that can be observed or measured

[x2 != 3] [x3 = 1, 3] v [x4 < 4]
=> [decision = A]

(each [...] is called a “selector”; the first one is TRUE if x2 is not equal to 3)

0.9 ([x1 = 3] [x3 >= 2])
+ 0.1 ([x3 = 2..4])
=> [decision = B]

(90% of the support comes from 2 selectors, and 10% from another)

A sample PLANT/ds rule:

0.8 * ([time = Aug..Sept]
[precip = 0.8]
[fruiting bodies = present]
[stem cankers = above 2nd node]
[fruit pods = absent])
+
0.2 * ([temp >= normal]
[canker lesion color = brown]
[# years crop repeated = 2 yrs.])

=> [diagnosis = diaporthe stem canker]

Rule evaluation:

- each selector [var = value] contributes a strength of evidence from 0 to 1
- if selector is not matched exactly, and is for a variable with linearly ordered domain, then a normal distribution is assumed
- for <u>conjunctions</u>, evidence is combined by product, min, or average
- for <u>disjunctions</u>, evidence is combined by max or (w1+w2-w1w2)
- evidence weights are scaled by module coefficients and added to get strength of evidence of conclusion
- The rule that scores best wins; Rules coming within 25% of best are considered alternatives
  Rules below 0.8 are not reported

PLANT/ds session

Each question which follows will have a list of answers. Choose the best answer, type the corresponding letter, then press return.

TIME OF OCCURRENCE?

A. April
B. May
C. June
D. July
E. August
F. September
G. October
H. Do not know

** B

CONDITION OF LEAVES?

A. Normal
B. Abnormal
C. Do not know

** B

CONDITION OF STEM?

A. Normal
B. Abnormal
C. Do not know

** why

IF[CONDITION OF STEM = NORMAL] then the following are indicated:

Downy Mildew
Bacterial Pustule
Phyllosticta Leaf Spot
Alternaria Leaf Spot

IF[CONDITION OF STEM = ABNORMAL] then the following are indicated:

Charcoal Rot
Rhizoctonia Root Rot
Phytophthora Rot
Brown Stem Rot
Anthracnose
Frog Eye Leaf Spot

CONDITION OF STEM?

A. Normal
B. Abnormal
C. Do not know

** A

CONDITION OF FRUIT-PODS?
A. Normal
B. Diseased
C. Few or none present
D. Does not apply
E. Do not know
** A

CONDITION OF SEED?
A. Normal
B. Abnormal
C. Do not know
** A

CONDITION OF ROOTS?
A. Normal
B. Abnormal
C. Galls or cysts present
D. Do not know
** A
.
.
.

CROPPING HISTORY?

A. Crop different than last year
B. Crop same as last year
C. Crop same for last three years
D. Crop same for four or more years
E. Do not know

** A

THE EVIDENCE PRESENTED SUGGESTS

BROWN SPOT WITH A DEGREE OF CONFIDENCE 1.00

ALTERNATIVELY

PHYLLOSTICTA LEAF SPOT WITH A DEGREE OF CONFIDENCE 0.82

The burden of the Knowledge Engineer

skill	*system*	*rules*
diagnosing soybean diseases	PLANT/ds	25
identifying bacteria	MYCIN	400
finding structure of organic compounds	DENDRAL	445
playing grandmaster chess	human	30,000
processing a visual scene	human	???

So many rules!

Human experts are not very good at writing rules

What if the computer could ***learn*** *its own rules?!*

This was tried with PLANT/ds (Michalski & Chilausky, 1981, Illinois)

Rules for diagnosing soybean diseases were generated from examples that were correctly classified by disease type by a human expert

Surprise! *(not really)*

Machine-derived rules performed better than the rules given by the human expert

original human rules	83% correct
improved human rules	93%
machine-derived rules	99%

How did the machine "learn" the correct rules?

data was collected for 350 sick plants thought to suffer from one of 15 diseases

the plant expert characterized each diseased plant using 35 different features
(each plant was represented as a point in a 35-dimensional space)

the plant expert divided the 350 data points into 15 different classes (one class per disease)

an inductive learning program generalized from the given points to find simple rules to describe each class
(this is called learning from examples)

the 15 rules (one rule for each class) were put into the knowledge base

these rules were tested using new cases of diseased plants

We could say that the machine "acquired knowledge" by examining the given examples

The system "learns" the necessary rules by performing inductive inference (generalization) over sets of examples

Machine learning is important for building large-scale expert systems

Knowledge-based systems: summary

Knowledge-based systems are ways to capture and use the knowledge of human experts

Knowledge-based systems need a knowledge base and a reasoning mechanism

IF-THEN rules are common, but other knowledge-representations are possible (eg., semantic nets)

Machine learning methods can help with large knowledge bases

More commercial successes here than any other part of AI

Knowledge-based systems: limitations

Knowledge-base generation and maintenance are difficult chores

Knowledge-based systems "know" only the things in the knowledge base

They do not know how their rules were developed

They do not know when to break their own rules

They do not look at problems from different perspectives

Most cannot reason at multiple levels

They typically cannot learn from their own experiences