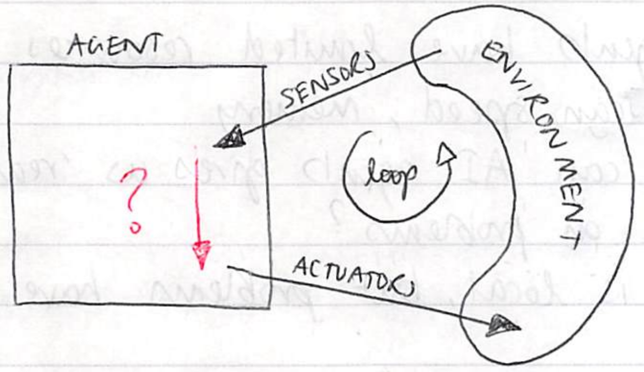
# Foundations Artificial Intelligence

* Introduction to Artificial Intelligence

Outline:
- Agent & Environment
- Applications of AI
- AI and Uncertainty

* Intelligent Agent

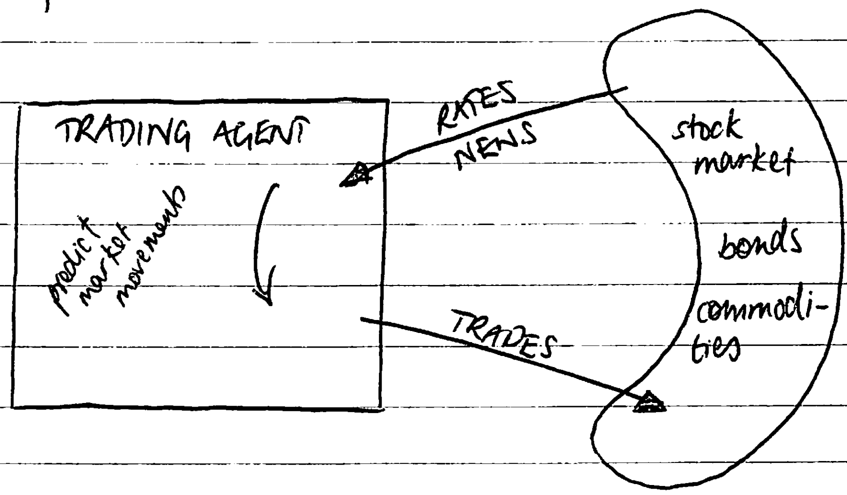



PERCEPTION ACTION CYCLE

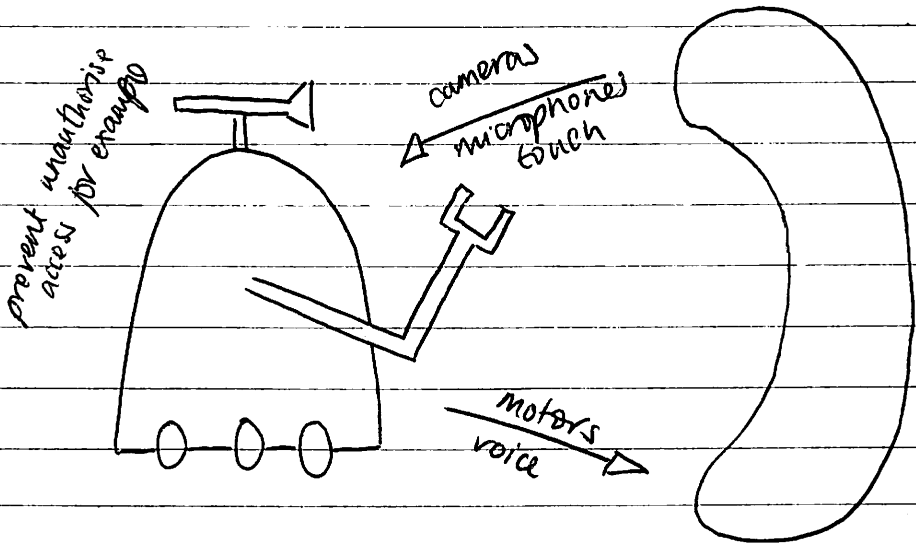
* Applications of AI

AI has sucessfully been used in
- Finance
- Robotics
- Games
- Medicine
- The web

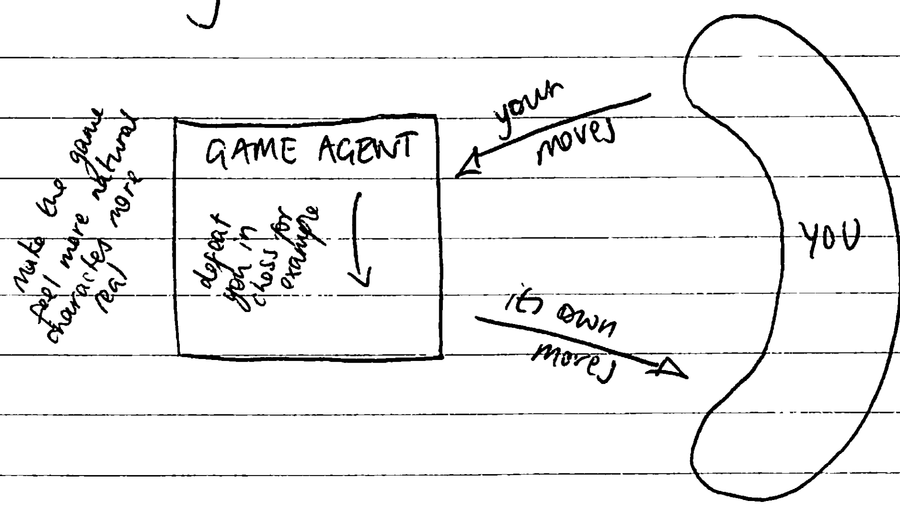
## AI in finance:



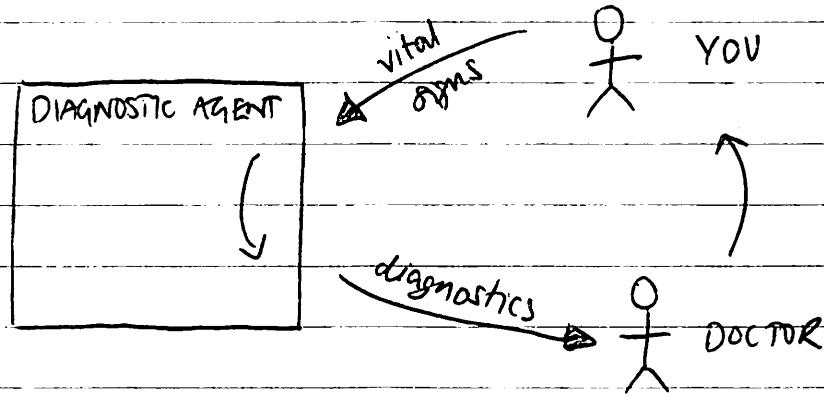

## AI in robotics



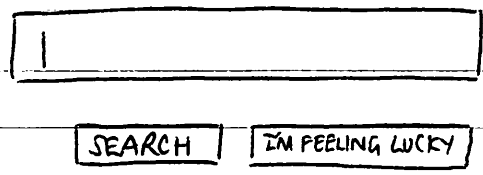

## AI in games



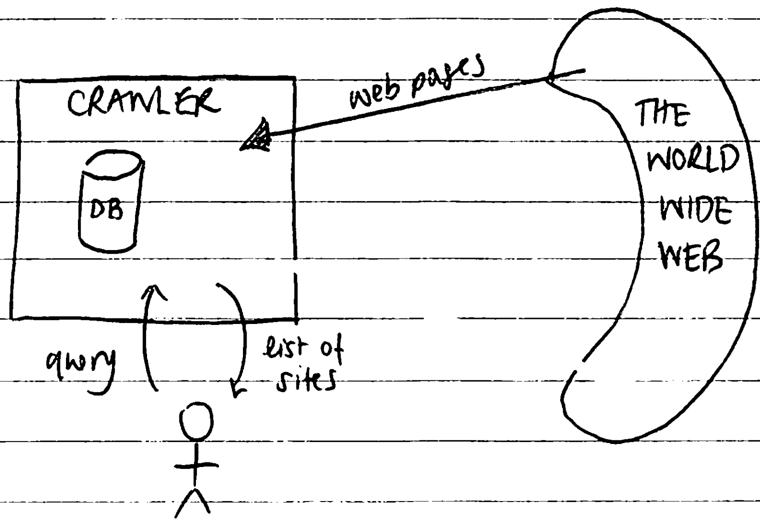
## AI in medicine




## AI and the web




find most relevant web pages for your search

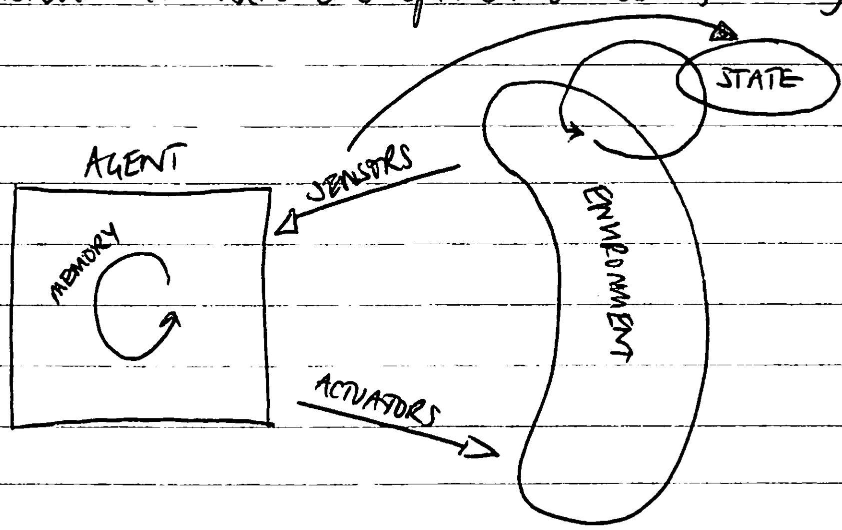


## Terminology

* FULLY vs PARTIALLY observable environment

requiring memory of past moves

what the agent can sense in any point in time is sufficient to make the optimal choice e.g. seeing all the cards on the table in a game




* DETERMINISTIC vs STOCHASTIC evironment

* DISCRETE vs CONTINUOUS environments

* BENIGN vs ADVERSARIAL evironments

BENIGN: stochastic

ADVERSARIAL: games where there are openents

Examples :

| | PARTIALLY OBSERVABLE | STOCHASTIC | CONTINUOUS | ADVERSARIAL |
|---|---|---|---|---|
| Checkers | | | | X |
| Poker | X | X | | X |
| Robot Car | X<br>momentary sensors is limited | X<br>other cars are unpredictable | X | X<br>other drivers |

AI as uncertainty management

AI = what to do when you don't know what to do?

Reasons for uncertainty
- Sensor limits
- Adversaries
- Stochastic environments
- Laziness
- ignorance

AI is a discipline which deals with uncertainty and manages it in decision making

Examples of AI

Language translation e.g. news articles - this is done by examining many articles which are published in both languages. These are used to find the most <u>probable</u> tra slation.
Example of its use in a chinese/english menu

Unit 1 summary
→ Key appuication
→ Inteligent agents
→ key attributes of environments
→ Sources of uncertainty
→ Rationality

* Knowledge-Based Artificial Intelligence

Overview :
- Conundrums in AI
- Characteristics of AI Problems & Agents
- AI in practice : Watson
- What is knowledge-based AI
- The four schools of AI

* Fundamental Conundrums of Artificial Intelligence

- Intelligent agents have limited resources
  - e.g. processing speed, memory
  - How then can AI agents give us 'real time performance on problems ?
- Computation is local, but problems have global constraints
- Logic is deductive, but many problems are not
  - How can AI agents address inductive and abductive problems ?
- The world is dynamic and knowledge is limited
  - How can AI agents address new problems ?
- Problem solving, learning and reasoning are complex, xplanation and justification even more so
  - How can we get AI Agents to explain or justify its decisions ?

* Characteristics of AI Problems

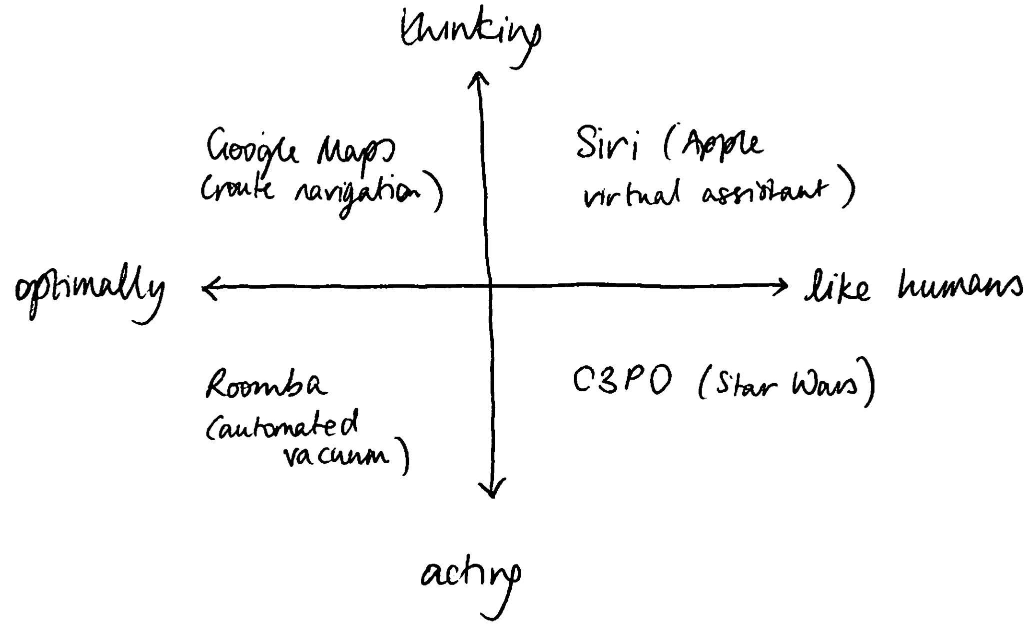
- Knowledge often arrives incrementally
- Problems exhibit recurring patterns
- Problems have multiple levels of granularity
- Many problems are computationally intractable
- The world is dynamic, knowledge is relatively static
- The world is open-ended but knowledge is limited

* Exercise: What is KBAI

Where does an autonomous vehicle fall on the spectrum?

Autonomous vehicles are in the acting optimally quadrant

* Exercise: The four schools of AI




* Problem Solving

Outline:
- What is a problem?
- Example: Route finding
- State spaces

# Unit 2: Problem Solving

Theory and technology of building agents to plan ahead and solve problems

Definition of a problem
- Initial state
- Actions(s) → {$a_1, a_2, a_3, \dots$} could be state dependent
- Result(s, a) → s'
- GoalTest(s) → true/false
- PathCost($s \xrightarrow{a} s \xrightarrow{a} s$) → n (the cost)

  StepCost(s, a, s') → n

Example: Driving from one city to another convert map of roads to a graph move along edges - you have the frontier, explored and unexplored areas

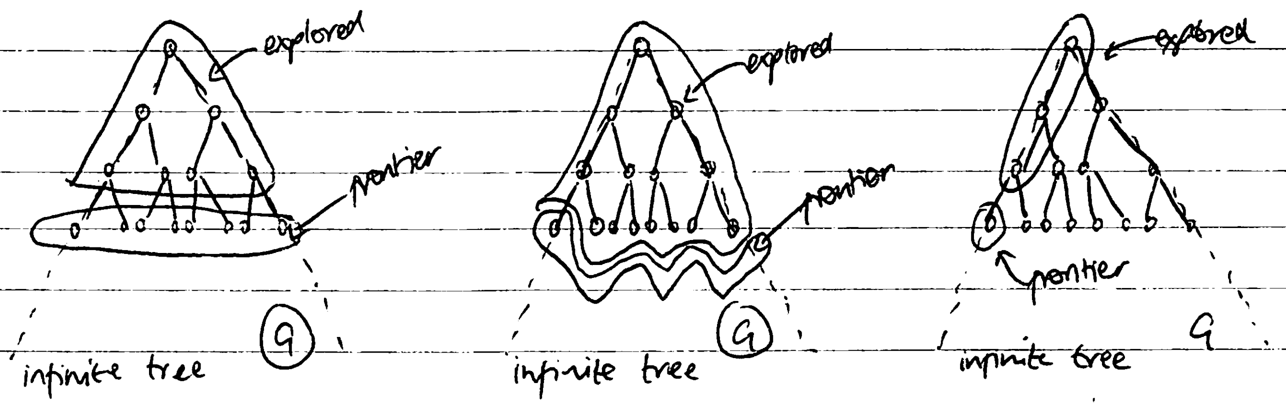
```
Function : TREE.SEARCH (problem):
    frontier = {[initial]}
    loop:
        if frontier is empty: return FAIL
        path = remove_choice(frontier)
        s = path.end
        if s is a goal: return path
        for a in actions:
            add [path + a -> Result(s,a)]
            to frontier
```

In general the goal test is applied when we remove the path from the frontier not when its added to the frontier

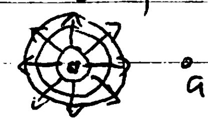
- Breadth first search: explore the shortest paths first (where edges are all 1)
- Cheapest first search: explore the cheapest path first (where the edges have assigned costs)

- Depth first search: explore the longest path first (where edges are all 1)

BREADTH FIRST SEARCH | CHEAPEST FIRST SEARCH | DEPTH FIRST SEARCH



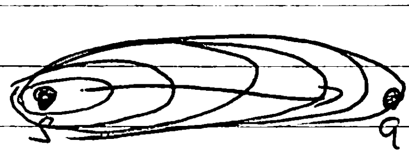

(Breadth first search and cheapest first search:)
- complete – guaranteed to find the goal if its at some finite place in an infinite tree
- search area expand uniformly in all directions i.e.

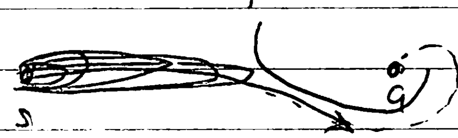



(Depth first search:) cheaper storage costs frontier and explored only contain n nodes

* Greedy-best first search → uses estimated distance from goal to decide which direction to expand its search in. This is good but doesn't find the optimal if there are barriers in the search paths

The idea is we want to get the optimal solution faster!

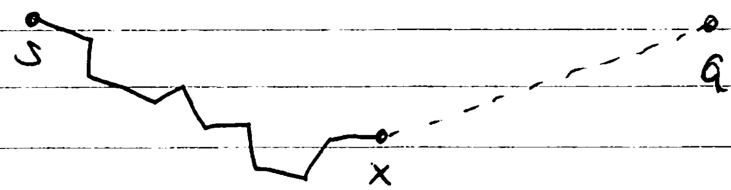


A* search : Best estimated total path cost first

expanding the paths which have the minimum value of the function $f$

$$f = g + h$$
$$g(\text{path}) = \text{path cost}$$
$$h(\text{path}) = h(s) = \text{estimated distance to goal}$$




A* finds the lowest cost path if :

- $h(s) \leq$ true cost

i.e.
- h never overestimates
- h is optimistic
- h is admissable

* Sliding blocks puzzel / 15 puzzel

| | | | |
|---|---|---|---|
| 1 | 2 | 3 | 4 |
| 5 | 6 | 7 | 8 |
| 9 | 11 | | 12 |
| 13 | 10 | 14 | 15 |

heuristics

$h_1$ = # misplaced blocks

$h_2$ = sum(distances of blocks) from correct posn

both these heuristics are admissable

$h_1 \leq$ # moves required fix it

$h_2 \leq$ # " " " "

$h_2 \geq h_1$ so $h_2$ is a better heuristic

A block can move $A \to B$
if (A adjacent to B) and (B is blank)

removing this gives $h_2$

removing both conditions gives $h_1$

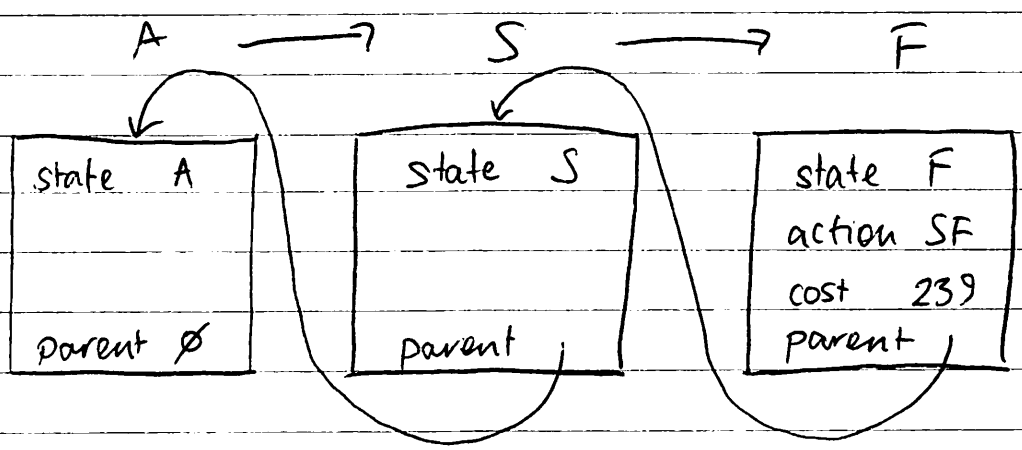
Take $h = \max(h_1, h_2)$

heuristic calculation can have a cost too.

removing condition only relaxes the problem so resulting heuristics can only underestimate the cost

This illustrates how a algorithm can be devised to come up with the estimated cost heuristic

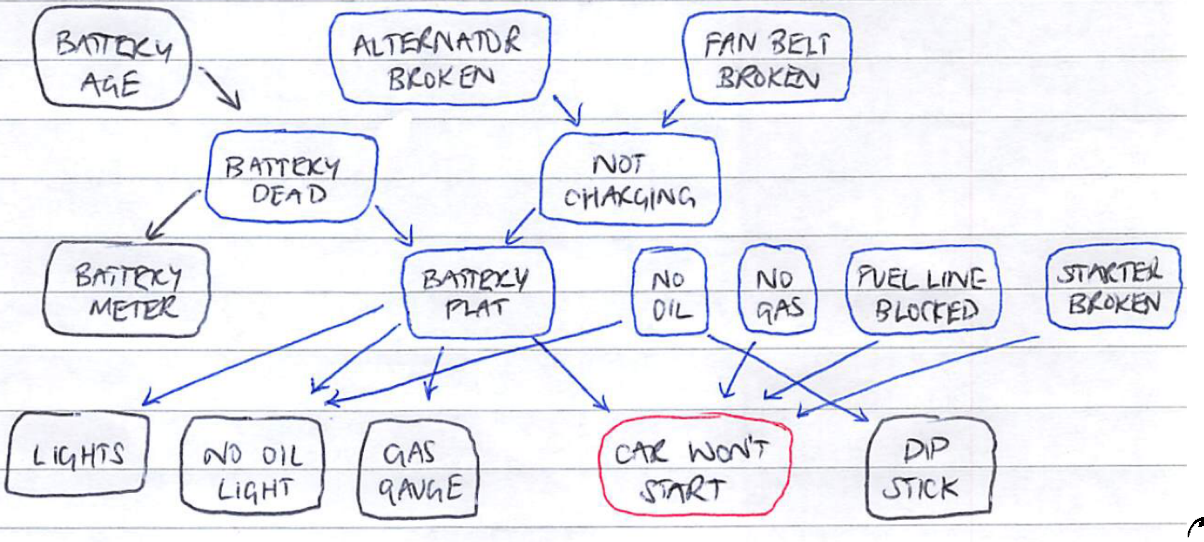
Problem-solving technology works when...
- Fully observable
- Known domain -
- Discrete
- Deterministic
- Static - no external influences on the environment



# * Probability in AI

Outline:
- Example: Bayes Network
- Conditional Probability
- Total Probability
- Joint Probability
- Bayes Rule

## Bayes Networks




- Bayes network assists you in reasoning from observable variables
- Child node is influenced by the Parent in a non-deterministic way

* This course:
  - → Binary events
  - → Probability
  - → Simple Bayes Networks
  - → Conditional independence
  - → Bayes Networks
  - → D-separation
  - → Parameter Counts
  - → Inference

* Bayes Networks are used in
  - → Diagnostics
  - → Prediction
  - → Machine Learning (finance, Google, Robotics)
  - → Particle filters
  - → Hidden Markov Models (HMM)
  - → MDP / POMDPs
  - → Kalman filters

* Probabilities:

$$P(A) = p \qquad P(\neg A) = 1-p$$

Independence

$$X \perp Y \Rightarrow \underbrace{P(X)P(Y)}_{\text{marginals}} = \underbrace{P(X,Y)}_{\text{joint}}$$

Dependence

$$P(Y) = \sum_i P(Y \mid X=i)\,P(X=i) \qquad \text{Total Probability}$$

$$P(\neg X \mid Y) = 1 - P(X \mid Y)$$

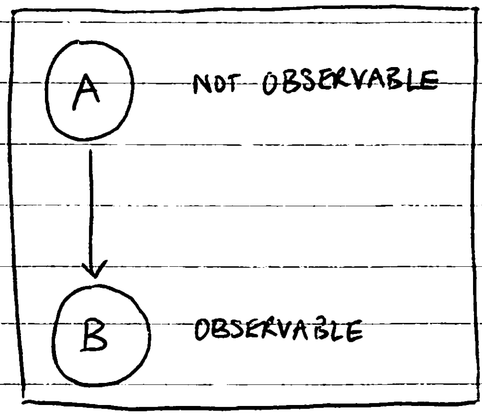
but $P(X \mid \neg Y) \neq 1 - P(X \mid Y)$

## * Bayes Rule

$$\underset{\text{POSTERIOR}}{P(A|B)} = \frac{\overset{\text{LIKELIHOOD}}{P(B|A)}\; \overset{\text{PRIOR}}{P(A)}}{\underset{\text{MARGINAL LIKELIHOOD}}{P(B)}}$$

$$P(B) = \underbrace{\sum_a P(B|A=a)\,P(A=a)}_{\text{total probability}}$$

Here B is the evidence ~~that A has happened or~~ is the case




Bayes Network

$P(A)$

$P(B|A)$
$P(B|\neg A)$

Diagnostic Reasoning : $P(A|B)$
$P(A|\neg B)$

## * More complex Bayes Networks

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)} \qquad P(\neg A|B) = \frac{P(B|\neg A)\;P(\neg A)}{P(B)}$$

$$P(A|B) + P(\neg A|B) = 1$$

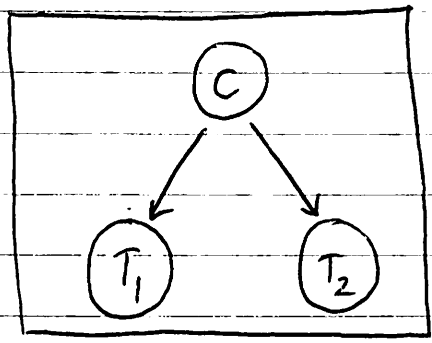
$$P'(A|B) = P(B|A)\,P(A) \qquad P'(\neg A|B) = P(B|\neg A)\,P(\neg A)$$

$$P(A|B) = \eta\, P'(A|B) \qquad P(\neg A|B)\ \eta\, P'(\neg A|B)$$

$$\eta = \left[P'(A|B) + P'(\neg A|B)\right]^{-1}$$

P' is called a 'pseudo probability'

* Conditional Independence

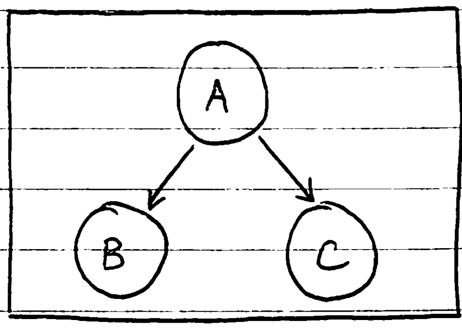



Bayes Network

Now let's assume the test for cancer is taken twice

* Conditional Independence

$$P(T_2 \mid C \cap T_1) = P(T_2 \mid C)$$




Given A
$B \perp C$
i.e. $B \perp C \mid A$

Note: $B \perp C \mid A \not\Rightarrow B \perp C$

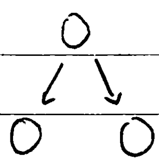
and

$B \perp C \not\Rightarrow B \perp C \mid A$

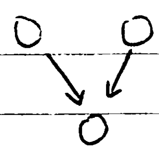
i.e. here we can show $S \perp R$ but $S \perp R \mid H$ is not true.

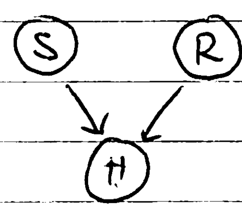
EXPLAIN AWAY CASE

Different type of Bayes Network



now:

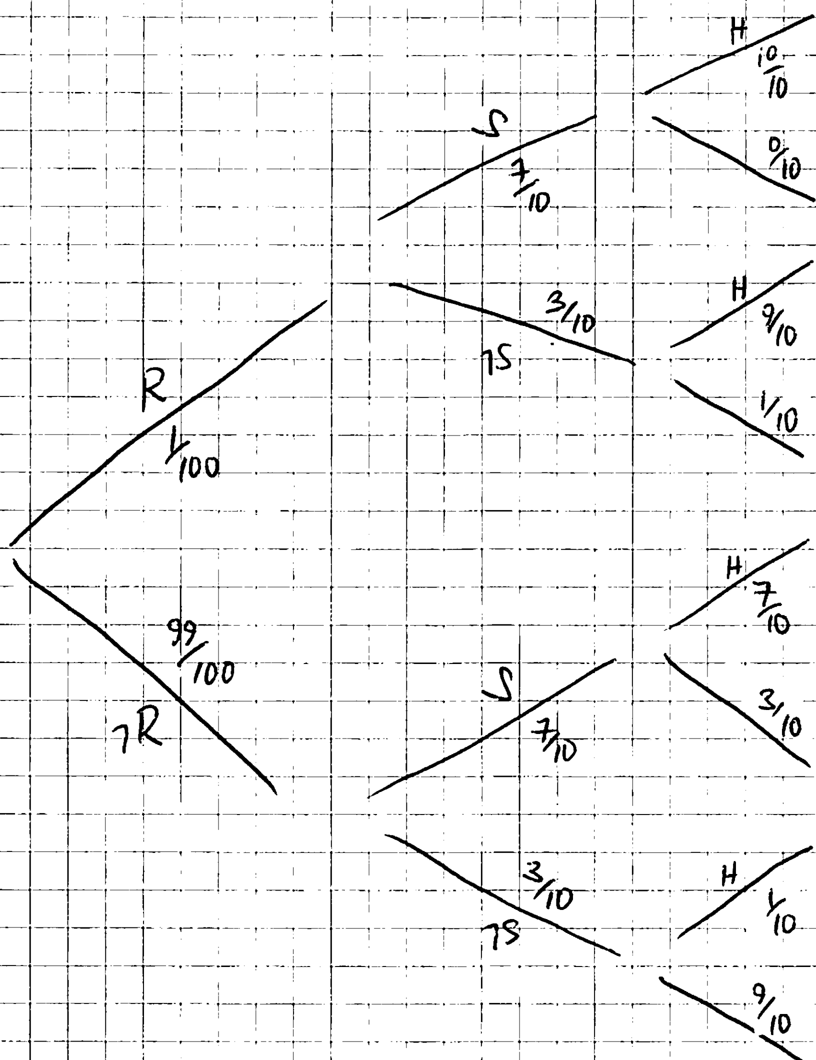





$P(S) = P(\text{sunny}) = 7/10$
$P(R) = P(\text{raise}) = 1/100$
$P(H) = P(\text{happy}) \longrightarrow$

$$\begin{cases} P(H \mid R \cap S) = 1 \\ P(H \mid R \cap \neg S) = 9/10 \\ P(H \mid \neg R \cap S) = 7/10 \\ P(H \mid \neg R \cap \neg S) = 1/10 \end{cases}$$

Example.




| | | | |
|---|---|---|---|
| $\frac{1}{100}$ | $\frac{7}{10}$ | $\frac{10}{10}$ | 70 |
| $\frac{1}{100}$ | $\frac{7}{10}$ | $\frac{0}{10}$ | 0 |
| $\frac{1}{100}$ | $\frac{3}{10}$ | $\frac{9}{10}$ | 27 |
| $\frac{1}{100}$ | $\frac{3}{10}$ | $\frac{1}{10}$ | 3 |
| $\frac{99}{100}$ | $\frac{7}{10}$ | $\frac{7}{10}$ | 4851 |
| $\frac{99}{100}$ | $\frac{7}{10}$ | $\frac{3}{10}$ | 2079 |
| $\frac{99}{100}$ | $\frac{3}{10}$ | $\frac{1}{10}$ | 297 |
| $\frac{99}{100}$ | $\frac{3}{10}$ | $\frac{9}{10}$ | 2673 |

$$P(R \mid H \wedge S) = \frac{P(R \wedge (H \wedge S))}{P(R \wedge (H \wedge S)) + P(\neg R \wedge (H \wedge S))}$$

$$= \frac{70}{70 + 4851}$$

$$= \frac{70}{4921}$$

Bayes formula $\Rightarrow$ $\dfrac{P(H \wedge S \mid R)\, P(R)}{P(H \wedge S)}$

$$= \frac{\frac{70}{100} \times \frac{1}{100}}{\frac{70}{10000} + \frac{4851}{10000}} = \frac{70}{4921} \approx \frac{1}{703}$$

* Characteristics of AI Agents

- Agents have limited computing power
- Agents have limited sensors
- Agents have limited attention
- Computational logic is fundamentally deductive
- AI agents' knowledge is incomplete relative to the world

* Exercise: What are AI problems?

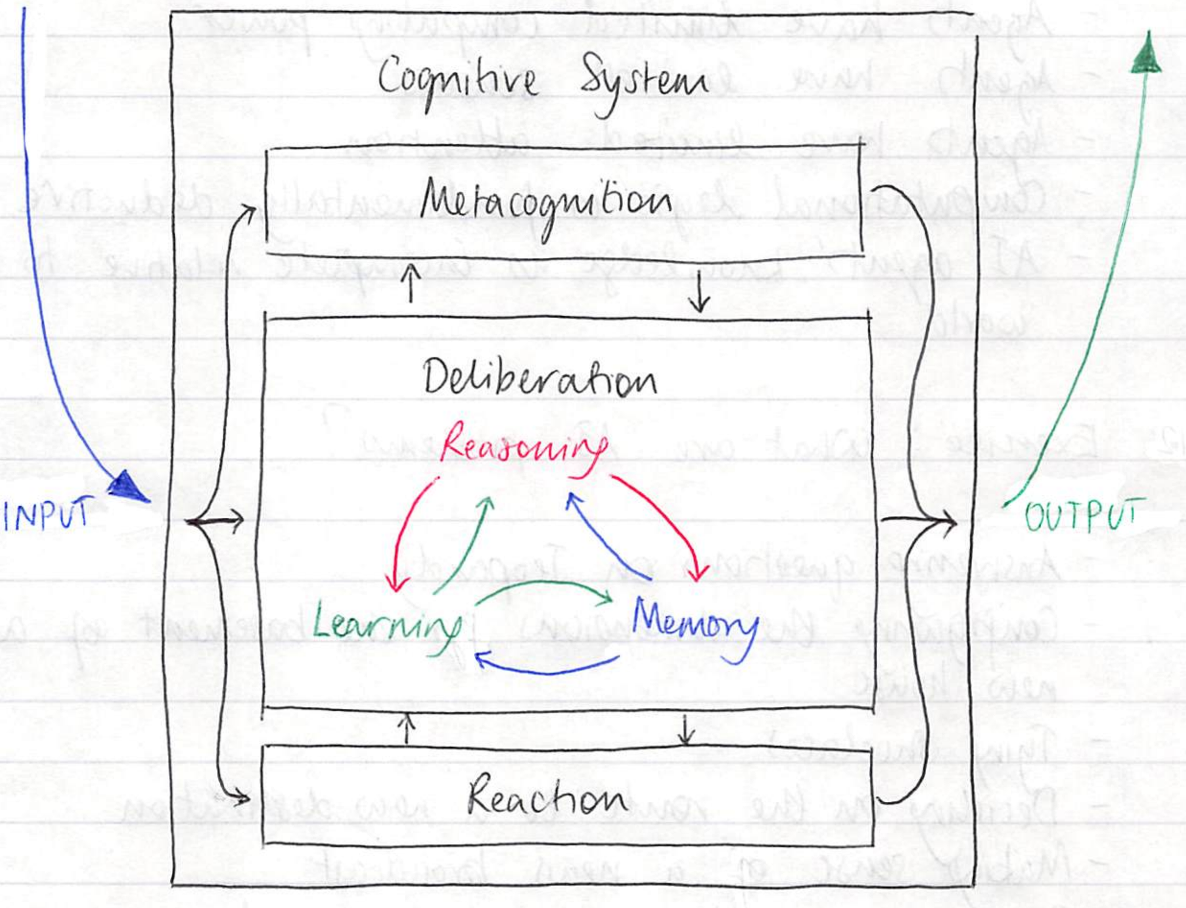
- Answering questions on Jeopardy
- Configuring the dimension for the basement of a new house
- Tying shoelaces
- Deciding on the route to a new destination
- Making sense of a news broadcast
- Designing a robot that walks on water
- Establishing whether a flower pot can be used as a drinking cup
- Deciding whether or not a new animal is a bird

An AI agent could be designed to solve any problem that a human can.

* AI in Practice: Watson (IBM's Jeopardy player)

Watson needs to be able to:
- Read clue
- Search knowledge base
- Decide on an answer
- Phrase the answer as a question.

* What is Knowledge-Based AI?




* Foundations: The four schools of AI