Unit - 4

Q1. Planning

- Planning is the task of coming up with a sequence of activities that will achieve a goal.

- Environments that are fully observable, static, finite, deterministic and discrete are called classic planning environments.

- states, actions and goals are used to represent a planning problem

Two types of planning in AI

- Forward State Space Planning
- Backward State Space Planning

- <u>Forward State Space Planning (FSSP)</u> : →

→ Behaves in the same way as forward state space search

→ Given an initial state S, we perform some necessary actions to obtain a new state S' and this is called progression.

→ This continues until we reach the target position.

eg. Disadvantage : Large branching factor
Advantage : The algorithm is sound.

- **Backward State Space Planning (BSSP) →**

→ Behaves in the same way as backward state space search.

→ In this, we move from the target state g to the sub goal g, tracing the previous action to achieve that goal. This process is called regression.

eg. Disadvantage: Not a sound algorithm, sometimes inconsistencies can be found

Advantage: Small branching factor (much smaller than FSSP)

Planning in artificial intelligence is about decision making actions performed by robots or computer programs to achieve a specific goal.

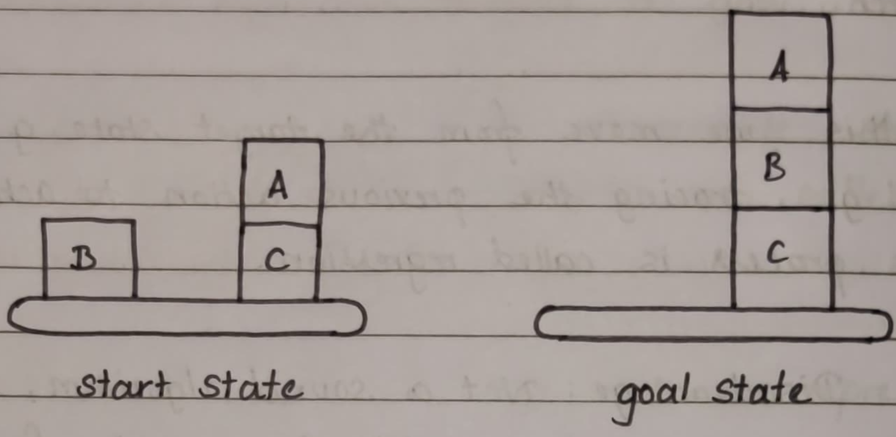
Execution of the plan is about choosing a sequence of tasks with high probability of accomplishing a specific goal.

Example:

The blocks world

- In the block world problem, three blocks labelled 'A', 'B' and 'C' are allowed to rest on a flat surface.

- The given conditions is that only one block can be moved at a time to achieve the target



- The blocks can be stacked, ~~over~~ but only one block can fit directly on top of another

- A robot arm can pick up a block and move it to another position, either on top of the table or on top of another block

- The arm can only pick up one block at a time, so it cannot pick up a block that has another block on top of it.

- Let us consider all three blocks on the table

- Planning problem ~~Algorithm~~ for it is shown as follows →

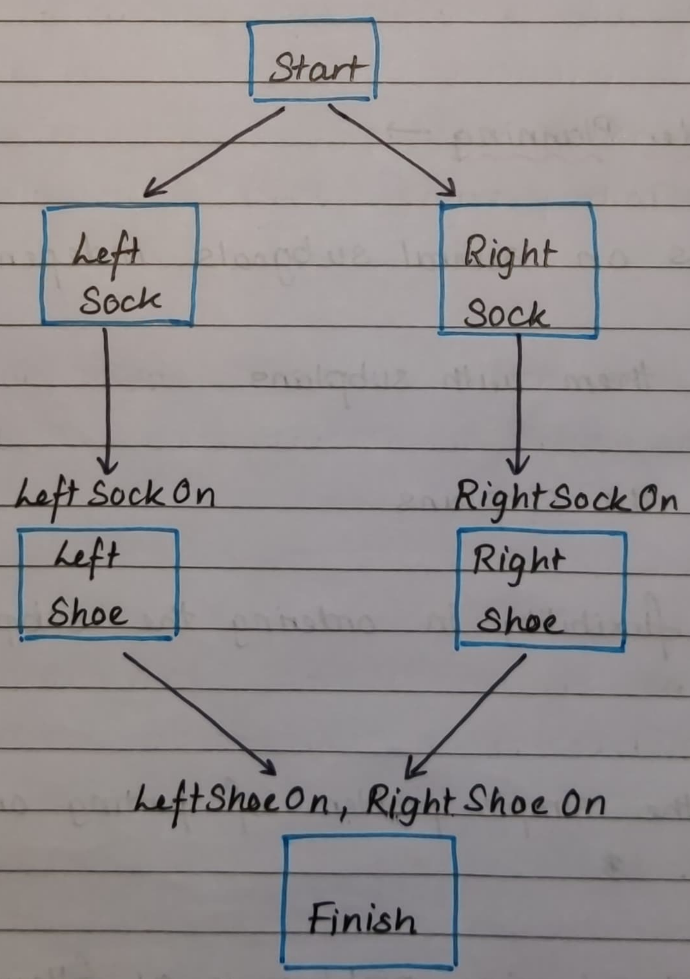
Init $(On(A, Table) \wedge On(B, Table) \wedge On(C, Table)$
$\wedge\ Block(A) \wedge Block(B) \wedge Block(C)$
$\wedge\ Clear(A) \wedge Clear(B) \wedge Clear(C))$
Goal $(On(A, B) \wedge On(B, C))$
Action $(Move(b, x, y)$,
PRECOND: $On(b, x) \wedge Clear(b) \wedge Clear(y) \wedge$
$Block(b) \wedge (b \neq x) \wedge (b \neq y) \wedge (x \neq y)$,
EFFECT: $On(b, y) \wedge Clear(x) \wedge \neg On(b, x) \wedge$
$\neg Clear(y))$
Action $(MoveToTable(b, x)$,
PRECOND: $On(b, x) \wedge Clear(b) \wedge Block(b) \wedge (b \neq x)$,
EFFECT: $On(b, Table) \wedge Clear(x) \wedge \neg On(b, x))$

Partial Order Planning →

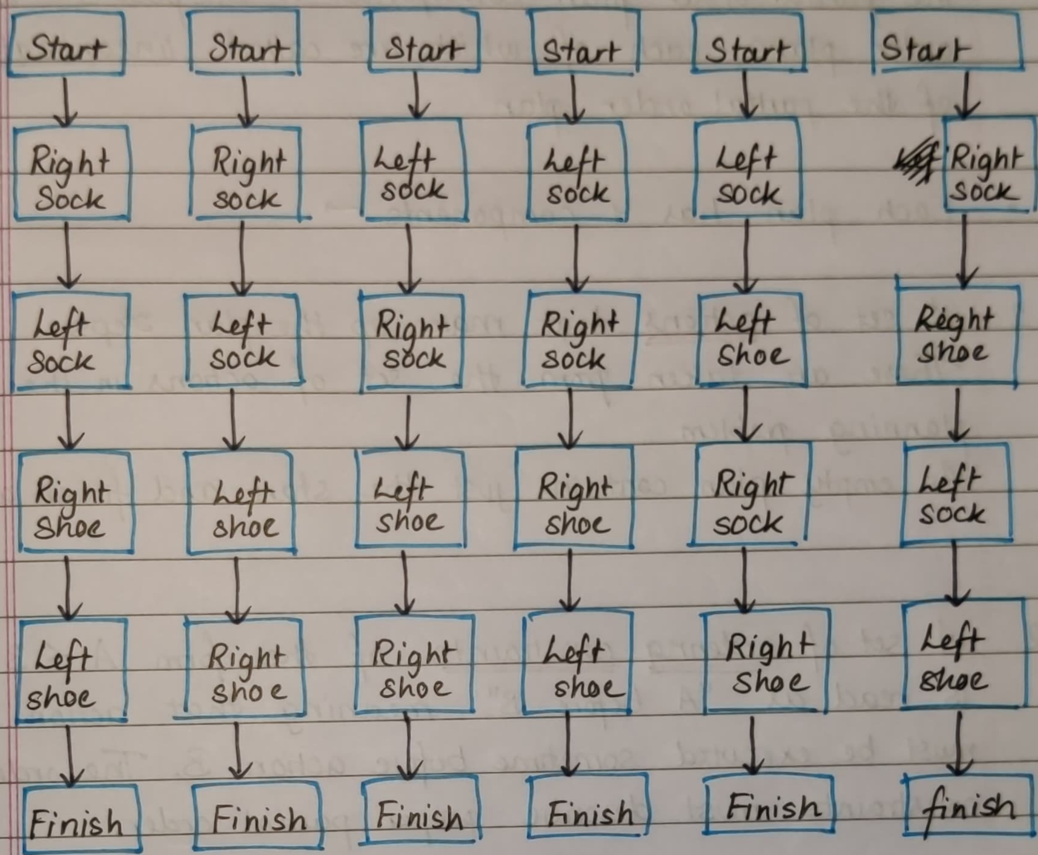
- POP works on several subgoals independently
- It solves them with subplans
- Combines the subplans
- There is flexibility in ordering the subplans
- Consider the simple problem of putting on a pair of shoes.
- The formal planning problem is as follows :-

Goal (RightShoeOn ∧ LeftShoeOn)
Init ()
Action (RightShoeOn ,
    PRECOND : RightSockOn , EFFECT : RightShoeOn)
Action (RightSockOn ,
    EFFECT : RightSockOn)
Action (LeftShoeOn ,
    PRECOND : LeftSockOn , EFFECT : LeftShoeOn)
Action (LeftSockOn ,
    EFFECT : LeftSockOn)

- Partial Order Plan →



- Total Order Plan →



- The planner that can place two actions into a plan without specifying which comes first is called a partial order planner.

- Here, the planner should be able to come up with a two action sequence RightSock followed by Right shoe forming the first conjunct of the goal and the sequence LeftSock followed by LeftShoe forming the second conjunct of the goal.

- These two sequences can then be combined to yield the final plan.

- The partial order plan corresponds to six possible total order plans, each of which are called linearization of the partial order plan

- Each plan has 4 components →

1. A set of actions that make up the plan steps. These are taken from the set of actions in the planning problem
   An empty plan contains just the start and finish actions.

2. A set of ordering constraints of the form $A < B$, which is read as "A before B" meaning that action A must be executed sometime before action B. The ordering constraints must describe proper partial order.

3. A set of casual links, written as $A \xrightarrow{P} B$, which is read as "A achieves p for B"

   eg. $\text{RightSock} \xrightarrow{\text{RightSockOn}} \text{RightShoe}$

   this asserts that RightSockOn is an effect of RightSock action and a precondition of RightShoe.

4. A set of open preconditions, a precondition is open if it is not achieved by some action Preconditions in the plan. Planners will work to reduce the no. of open preconditions to the empty set, without introducing a contradiction

Q2. Bayes Theorem

Probability of flipping a coin and getting heads →

$$P(\text{Head}) = \frac{\text{Heads side}}{\text{Heads side} + \text{Tails side}}$$

$$= \frac{1}{(1+1)} = \frac{1}{2} = 0.5 = \underline{\underline{50\%}}.$$

Probability of rolling a dice and getting a 3 →

$$P(\text{die} = 3) = \frac{1 \text{ side with a } 3}{\text{six sides total}}$$

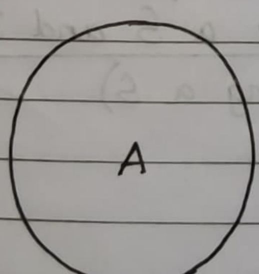
$$= \frac{1}{6} = 0.167 = \underline{\underline{16.7\%}}.$$

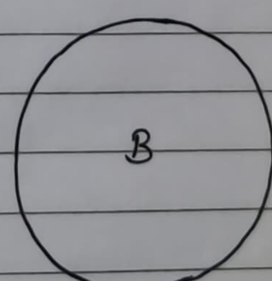
Conditional probability →

Events A and B are events that are not mutually exclusive, but occar conditionally on the occurence of one another.

probability that A will occur : $p(A)$

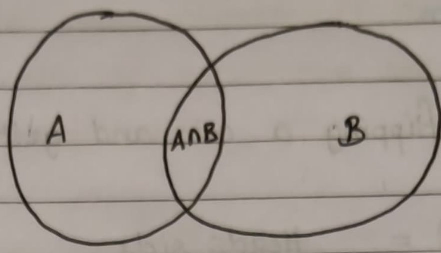
probability the B will occur : $p(B)$





The number of times both A and B occur or the probability that both the events occur is called the joint probability $p(A \cap B)$

Joint probability of A and B



The probability that event A will occur if event B occurs is called conditional probability.

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(A \text{ and } B)}{P(B)}$$

Suppose we have 2 dice and we want to know the probability of getting an 8. Normally the probability on rolling both the dice together would be 5/36.

But image we roll the first dice first and get a 5, now what is the probability of getting an 8?

There is only 1 way to get an 8 after a 5 has been rolled. You have to get a 3 on the second roll.

P(getting 8 using 2 die | given that we roll a 5 using the first dice)

$$= \frac{P(\text{getting a 5 and 3})}{P(\text{rolling a 5})}$$

$$= \frac{1/36}{1/6}$$

$$= \frac{6}{36} = \frac{1}{6}$$

Bayes theorem ⟶

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \quad , \quad P(A \cap B) = P(B \cap A)$$

then ~~and~~ $P(A \cap B) = P(B|A) * P(A)$

since $P(B \cap A) = P(B|A) * P(A)$

$$\therefore \boxed{P(A|B) = \frac{P(B|A) * P(A)}{P(B)}}$$

$$P(h|D) = \frac{P(D|h) * P(h)}{P(D)}$$

where $P(h)$ is the prior probability of the hypothesis
$P(D)$ is the prior probability of the data and
$P(h|D)$ is the posterior probability (probability that the hypothesis is true, given the data)

Q3. Naive Bayes Classifier

In this example, we want to use Bayes theorem to find out the likelihood of playing tennis for a given set of weather attributes

$f(x) \in v = (\text{yes}, \text{no})$ ie. $v = ($yes we will play tennis, or, no we will not play tennis$)$

the attribute values are $a_0 \cdots a_3 = ($Outlook, Temperature, Humidity, and Wind$)$

Day	Outlook	Temp	Humidity	Wind	Play tennis
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

We write our $f(x)$ in the form:

$$P(\text{Play tennis} \mid \text{Attributes}) = \frac{P(\text{Attributes} \mid \text{Play tennis}) * P(\text{play tennis})}{P(\text{attributes})}$$

OR

$$P(v \mid a) = \frac{P(a \mid v) * P(v)}{P(a)}$$

Using the table, let us find the overall probabilities and conditional probabilities →

$P(\text{play tennis} = \text{Yes}) = 9/14 = 0.64$

$P(\text{play tennis} = \text{No}) = 5/14 = 0.36$

Outlook:

$P(\text{Outlook} = \text{Sunny} \mid \text{Play tennis} = \text{Yes}) = 2/9 = 0.22$
$P(\text{Outlook} = \text{Sunny} \mid \text{Play tennis} = \text{No}) = 3/5 = 0.6$

$P(\text{Outlook} = \text{Overcast} \mid \text{play tennis} = \text{Yes}) = 4/9 = 0.44$
$P(\text{Outlook} = \text{Overcast} \mid \text{Play tennis} = \text{No}) = 0/5 = 0$

$P(\text{Outlook} = \text{Rain} \mid \text{Play tennis} = \text{Yes}) = 3/9 = 0.33$
$P(\text{Outlook} = \text{Rain} \mid \text{Play tennis} = \text{No}) = 2/5 = 0.40$

Temperature:

$P(\text{Temp} = \text{Hot} \mid \text{Play tennis} = \text{Yes}) = 2/9 = 0.22$
$P(\text{Temp} = \text{Hot} \mid \text{Play tennis} = \text{No}) = 2/5 = 0.40$

$P(temp = Mild \mid Play\ tennis = yes) = 4/9 = 0.44$
$P(temp = Mild \mid Play\ tennis = no) = 2/5 = 0.4$

$P(temp = cold \mid Play\ tennis = yes) = 3/9 = 0.33$
$P(temp = cold \mid Play\ tennis = no) = 1/5 = 0.2$

Humidity :

$P(humidity = high \mid Play\ tennis = yes) = 3/9 = 0.33$
$P(humidity = high \mid Play\ tennis = no) = 4/5 = 0.80$

$P(humidity = normal \mid Play\ tennis = yes) = 6/9 = 0.66$
$P(humidity = normal \mid Play\ tennis = no) = 1/5 = 0.20$

Wind :

$P(wind = weak \mid play\ tennis = yes) = 6/9 = 0.66$
$P(wind = weak \mid play\ tennis = no) = 2/5 = 0.40$

$P(wind = strong \mid play\ tennis = yes) = 3/9 = 0.33$
$P(wind = strong \mid play\ tennis = no) = 3/5 = 0.60$

Suppose the day is described by :

a = (Outlook = sunny, Temperature = cool, humidity = high, wind = strong)

What would our Naive Bayes classifier predict in terms of playing tennis on a day like this ?

Whichever eqn $P(Yes \mid (Sunny, cool, high, strong))$ or $P(no \mid (sunny, cool, high, strong))$ has the higher probability will be the prediction of the classifier.

$$P(yes \mid (sunny, cool, high, strong)) = \frac{P((sunny, cool, high, strong) \mid yes) * P(yes)}{P(sunny, cool, high, strong)}$$

noting that :

$$P(sunny, cool, high, strong) = P((sunny, cool, high, strong) \mid yes) + P((sunny, cool, high, strong) \mid no)$$

So our eqn expands to :

$$= \frac{P(sunny \mid yes) * P(cool \mid yes) * P(high \mid yes) * P(strong \mid yes) * P(yes)}{P((sunny, cool, high, strong) \mid yes) + P((sunny, cool, high, strong) \mid no)}$$

$$= \frac{(0.22 \times 0.33 \times 0.33 \times 0.33) * 0.64}{(0.22 \times 0.33 \times 0.33 \times 0.33) \times 0.64 + (0.6 \times 0.2 \times 0.8 \times 0.6) \times 0.36}$$

$$= \frac{0.0051}{0.0051 + 0.207}$$

$$= 0.1977.$$

Similarly -

$P(No|(Sunny, cool, high, strong)) =$

$$\frac{P((sunny, cool, high, strong)|no) * P(no)}{P(sunny, cool, high, strong)}$$

$$= \frac{0.207}{0.0051 + 0.207}$$

$$= \underline{\underline{0.8023}}$$

∴ The naive bayes classifier gives a value of just about 20% for playing tennis in the described conditions, and a val of 80% for not playing.

So the prediction is that tennis will not be played in such conditions

Q4. Belief Network

A Bayesian network / Bayes Network / Belief network represents a set of random variables and their conditional dependencies via a directed acyclic graph (DAG)

For example, a Bayesian network could represent the probablistic relationships between diseases and symptoms. Given symptoms, the network can be used to predict the presence of various diseases.

In Bayesian network DAG's, the nodes represent the random variables

The edges represent the conditional dependencies.

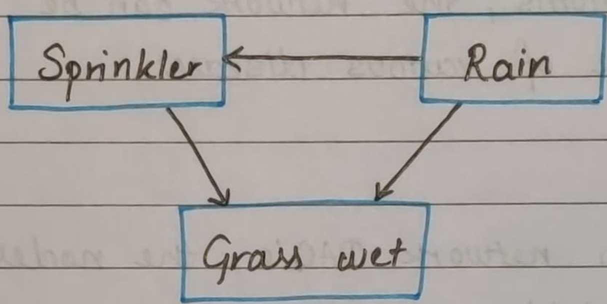
If there is no ~~co~~ connection between two nodes, it means that those variables are conditionally independent of eachother.

Each node is associated with a probability function which takes a particular set of values of the node's parent variables as input and gives the probability of the variable represented by the node as the output

Suppose there are two events which could cause grass to be wet, either sprinkler is on, or it is raining.

Also the rain has direct effect on the use of sprinkler; ie. when it rains, sprinkler is not turned on.

The situation can be modelled using the following Bayesian network.

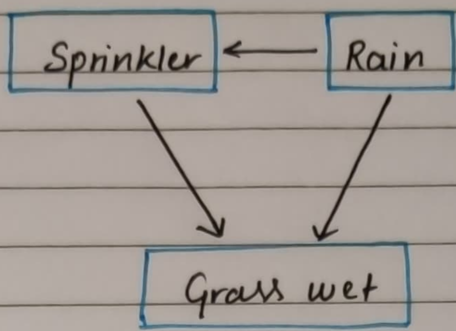


All three variables have two possible values, T (for true) and F (for false)

G = Grass wet (yes / no)
S = Sprinkler turned on (yes / no)
R = Raining (yes/no)

The model can answer questions like, "what is the probability that it is raining, given the grass is wet?" by using conditional probability formula

R	Sprinkler T	Sprinkler F
T	0.4	0.6
F	0.01	0.99



Rain T	Rain F
0.2	0.8

S	R	Grass wet T	Grass wet F
T	T	0.99	0.01
T	F	0.9	0.1
F	T	0.8	0.2
F	F	0.0	1.0

$$P(R=T, G=T, S=F) = P(R=T) * P(S=F \mid R=T) * P(G=T \mid S=F, R=T)$$
$$= 0.2 \times 0.6 \times 0.8$$
$$= \underline{\underline{0.096}}.$$

Using a Bayesian Network can save considerable amounts of memory, if the dependencies in the joint distribution are sparse

Bayesian networks are intuitively easier for a human to understand with a sparse set of direct dependencies and local distributions than complete joint distributions.