Richard Knight - book of AI

Python - 1989
Java - 1995

Came in picture

Artificial Intelligence :-

[Constraint satisfaction, DFS, BFS, Prolog], minmax Algo, α-β cutoff,

Most Imp ***

A*, AO*, best First, Hill climbing

*1) Approach to AI (Heuristic Search, Game playing)
[No basic level questions ?]

2) * Knowledge Representation [Approaches (Predicate logic), Reasoning] **
[forward & backward Reasoning]

3) * Planning (overview, Hierarchical, goal stack]

4) * NLP [Natural Lang. Processing] [Syntactic, Semantic]

5) ** Multiagent system [types, Properties].
*** [No basic level questions asked]

6) Fuzzy Sets [Crisp fuzzy set, α-cut, operations]

supervised & Unsup. Learning

7) ** ANN & Genetic Algo. (Single, Multilayer feed forward, ML]
Artificial Neural N/w

2018 Queat^ns. DBMS - topic - Big data. [What is Big data?]

1

* Introduction to AI :-

- AI Evolution - During World war-II [1939-1945]
  First Computer made - purpose to break the German Commn.

* Alen Turing - Major role in making m/c

* Alen-Turing published one paper -
  "Can Machines Think?"
  - Alen Turing.

19-50

* We want to make a m/c's like, behave like humans, think like humans, work like humans.

* In fact, we want to make m/c's ~~like~~ intelligent like human intelligence.

e.g. 1] Smart phones.

- In smart phones there is a very major important application, that is Google assistant, Siri, Alexa.

* In this application, u can ask queries to the application by speaking except typing.

* When u are speaking then that app is converting ur language through NLP - [Natural Lang. processing] & also it is giving answer to you

- In google data is not handling in Gbytes or Terabytes Google is handling Hexabytes, betabytes of data.

* - within the fraction of second or milisecond Google gives ur small questions answers, How?
⇒ Backword of this logic that is Algorithms of AI is working.

e.g. 2] Smart Cars:-

- Tesla compani made a smart car's as driverless.
- In US more than 50,000 care are running automatically.
- U have to just sit in car, just Give it Destination, { how to run, how to give break, or when obstacle in roads then how to stop, or change the Gear & how to race }

* There is no need to tell the smart cars or no need of driver.

- It can handle automatically.

* We want that machines also think like humans & make decisions like humans
- want m/c'ns learn & peecept & decision making like human beings.

Humans think, & learn, & then make a decisions
- We want to make decision making power to the macines.

* AI is trying to break the barrier or constraints in betn humans & Roborts.
- इसके पीछे, learning algorithms, searching algo, reasoning algo's are working behind this constraints.

e.g.] NETFLIX or YOUTUBE or AMAZONE or MEESHO

⇒ The recommendations for a customers.
means, whatever u have searched or liked according to that these apps gives u a reccomandations according to your choice.
Why? Because, At the backend, the AI is analyzing you completely.
- It will impact on your decisions also.
- means our complete future depending on AI.

e.g. 4] One man wants to order PIZZA:-
- He opened the application & entered the cust-id & he get options for pizz like - veg pizza or Non veg pizza - Cheez burst.
- And He selected cheeze burst.
- After selecting Cheeze burst pizza, He get a warning from the app that "Your Colesterol level value is High".
* - How he get this warning or message?
possibillity - through Mob. no.
means he gave in medical his mob. no. & through customer id & mobile no. matching he get the warning

— mean he also dnt know about his Colesterol but he will get a msg. or warning.

warning 1) — "Don't order this Pizza becz, ur Colesterol is high."
warning from the app.
— but he ignored the warning & ordered Pizza.
— Next time, Her also choosen the burstcheeze Pizza at that time He will get the warning with
warning 2) "the Colesterol value in Bold".
— Still he can ignore & order Pizza.
* — Next time Expert system of AI can remove the option cheeze burst from his cust-Id
— means Automatically cheeze burst option removed or deactivated.
* This expert system is a decision making ES on the basis of AI.

(2)

⇒ 1) Reasoning :-
means we hv to think logically or sensible.
e.g. u are giving any competitive exam.
— For qualifying that exam u want more than 40% marks
— And A student get 60% in that exam.
From this we can get logic or reasoning is that A Student get qualified in this compet^n exam.
— This is easy for us (human beings)
— We want to load this logic or reasoning into machines.

2] Learning :-
- We are learning the things & getting diff experiences & dnt making mistakes in future.
— We are expecting Learning from Machine's because, we want learning from m/c's & m/c will store that data & in future he can handle the same situations in better way.

3] Problem Solving :-

- suppose, we are taking game playing ; in that alot of game playing algorithms.
- In any mathematical ~~problem~~ computation, we are ~~solv~~ trying to solve the problem in maths.
- However, humans are solving problems from mind, we want that the m/c's also try to solve the problems by thinking from their mind., on that basis machines can make decision.

4] Perception :-

- Human's can guess the emotions from, eyes, nose, eyebrows, ears., or facial expressions.

* or humans can pecept from eyes, ~~ear~~ nose, ears. (organs) & he ~~to~~ make decision after perception.

we want this above perceptions from machines.

e.g. Smart Cars.

Tesla Company, provided a automated cars.

- suppose, any car sees the Jam on that road, then that car will be percepted & he will forword the messages to the further cars that "Don't go through this road.

* - Automatically this car will transmit this message to other cars.
- Here m/c's making self decision.

* AI & ML & Deep Learning, these topics are very important, now, a days..

* We want to make

③

* State Space

- AI's major a

* How to Solv

- In 1960, 70's

- Suppose, we c
e.g. Tic-
Ches
how,
m/c's

* In problem
very Imp
State space
no. of state
e.g. Let say
Start s
① How to re
② Analyze.

S : { S, A

S = Start
S = Tot.
A - Set
e.g. 8-p
we have 3x3

2	3	
5		
8	7	

start state

when
always
intern

* We want make a m/c's like human intelligence.

③

* "State Space Search" :-

- AI's major application - Problem Solving.

* How to solve a particular problem ?

- In 1960, 70's Era, major research in AI - On Problem Solving

- Suppose, we are talking related to game -
  e.g. Tic-Tac-Toe, Water jug problem, 8-queen problem
  chess game. etc.
  how, we can solve these problems through m/c's.

* In problem solving area, state space searching is very Imp.

State Space Search -
no. of states in which the problem can go.
e.g. Let say, we want reasearch on any topic

Start State $\xrightarrow{\text{Intermediate states}}$ Goal State

① How to represent the problem precisely. [Precise]
② Analyze.

S : { S, A, Action (s), Result (s,a), Cost (s,a) }

S = Start, Goal, intermediate states.
S = Tot. no. of states
A - Set of all possible actions.

e.g. 8-puzzle problem
we have 3x3 board, 9-places, 8 no. tiles. & only 1 empty tile.

Start State (S)

2	3	4
5		1
8	7	6

Goal State (G)

1	2	3
8		4
7	6	5

When we try to explore this problem, then always will compare with Goal State
intermediate states compare with

A → All possible actions

R →

*means we can move the tile or no's to up, down, left or right [4-possibilities

S

e.g.

2	3	4
5		1
8	7	6

3x3

1	2	3
8		4
7	6	5

2		4
5	3	1
8	7	6

2	3	4
5	7	1
8		6

2	3	4
	5	1
8	7	6

2	3	4
5	1	
8	7	6

~~Down~~ Upward move

Downward move

left

	2	4
5	3	1
8	7	6

2	4	
5	3	1
8	7	6

* Uninformed Search (Blind Search) Algorithms —

1] BFS — Breadth-First Search
2] DFS — Depth-First-Search
3] Depth-limited Search (DLS)
4] Iterative deepening depth-first search
5] Uniform cost search
6] Bidirectional Search

R — Result
C — Cost.

we want searching in minimum Cost but it gives an optimal soln

In AI there are 2 types of Searching

1] Uninformed search (Blind Search)

GOODLUCK Page No.
Date / /

* For our goal states we can search by 2 types Uninformed search & informed search.

- In uniformed search, we are comparing every intermediate state ~~to~~ with Goal states, thats why its searching exponential time is increased. [~~expone~~

* Informed Search - [Heuristic Search]

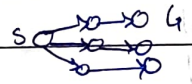
- It tries to solve the problem quickly

$\boxed{\text{Time} = O(b^d)}$ b - branching factor.
d - depth.

thats why we prefered to refer the informed searching.

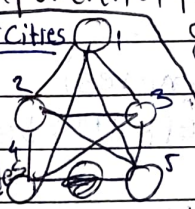
Difference between Uninformed & Informed Searching

Uninformed (Blind Searching)	(Heuristic) Informed
① Search without information	① Search with information
② No knowledge	② Use knowledge to find steps to Solution.
③ Time Consuming.	③ Quick Solution.
④ More Complexity (time, space)	④ Less Complexity (Time, Space)
⑤ DFS, BFS. et.	⑤ A*, Heuristic DFS, Best first Search
	Heuristic - h(n)
	- optimality soln not garanteed



Travelling Salesman problem in Exponential time growth in uninformed Search

e.g. TSP - 5 cities



$(n-1)!$

$(5-1)!$

$4! = 24$ states

- Non polinomial (NP) problems, time & space increased & cost also increased & gives optimal soln.

BFS - [Breadth First Search]

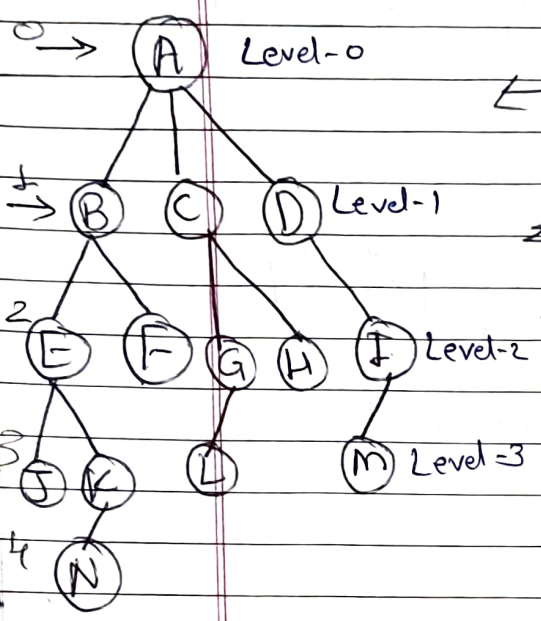
- Uninformed Search technique. [Blind]
- FIFO (queue)
- Shallowest Node
- Complete.
- Optimal
- Time Complexity. = $O(V+E)$ in BS, where V - no. of node, E - no. of Edges

queue (DS) & in AI space complexity - $O(b^d)$

b - branch factor

d - depth.

d = 4



~~A~~

~~BCD~~

~~CDEF~~

~~DEFGH~~

~~EFGHI~~

~~FGHIJK~~

~~GHIJK~~

~~HIJKL~~

~~IJKL~~

~~JKLM~~

~~KLM~~

~~KMN~~

~~MN~~

N

Suppose, we want to search G node in this tree then we can write path as $\boxed{ACG}$

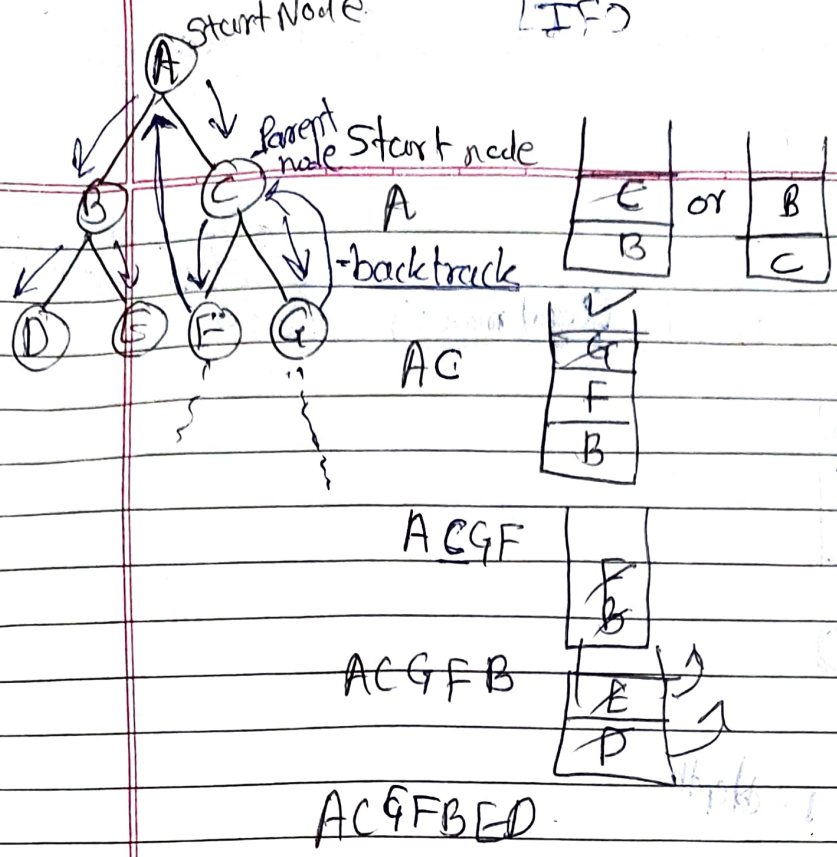
path for searching G is $\boxed{ACG}$ this is a goal state.

b=3, & d=2 $O(b^d) = O(3^2) = 9$ search.

DFS - Depth First Search (DFS):-

- Uninformed search technique
- Stack (LIFO)
- Deepest Node
- Incomplete - (It will not give result may be) (Infinite moves)
- Non Optimal
- Time Complexity - $O(V+E)$ V - no. of nodes, E - no. of edges

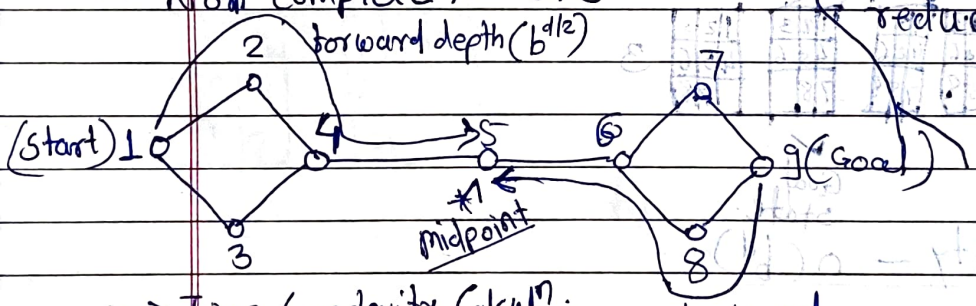
$O(b^d)$



* Bidirectional Search :-

- Two simultaneous search from an initial node to goal & backword from goal to initial node, stopping when two meet.
- Time Complexity : $2(b^{d/2})$
- Complete in BFS
- Not complete in DFS

we use the bidirectional search except BFS & DFS bcz, time complexity is reduced.

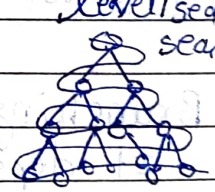


⟹ Time Complexity Calcul^n :-

∴ $\underset{\text{depth}}{\text{forw}} + \underset{\text{depth}}{\text{back}} = O(b^{d/2} + b^{d/2})$

$\boxed{\text{Time Complexity} = 2b^{d/2}}$ & In worst case $\boxed{O(b^d)}$

levelised search



① Because, in BFS there is levelized search like so, it is Complete bidirectional.

② In DFS, search is from depth to backtrack nodes like Parallely S ⟶ G, S ⟵ G there is a no chance to meet each other. So, it is not complete search.

BFS

DFS

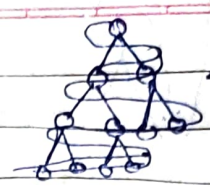
* 8-puzzle Problem without Heuristic (root force method)

→ [Uninformed / Blind search]

→ BFS

→ $O(b^d)$

→ 4 moves (Up, down, left, right) (legal moves)



BFS levetise

S

1	2	3
	4	6
7	5	8

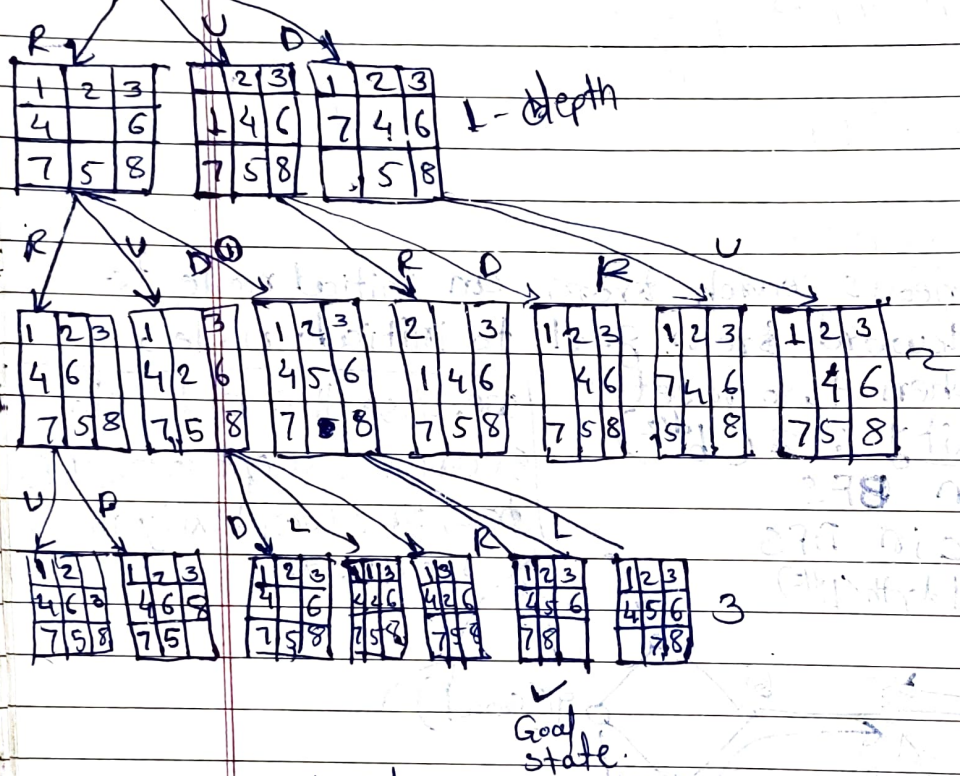
Tile

(Start State)

G

1	2	3
4	5	6
7	8	

(Goal State)

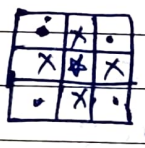


Time complexity – $O(b^d)$

In worst Case – $b = (3^{20})$

* Branching factor (b):-

It depends on empty factors position –



$\underset{\text{posit}^n}{4} \times \underset{\text{moves possible}}{2} + 4 \times 3 + 1 \times 4 / 9 = \frac{8+12+4}{9} = \frac{24}{9} = 2.6 = \underset{\text{value upper}}{\underline{\underline{3}}}$

$\therefore O(b^d) = O(3^3) = O(27)$

* Heuristic in AI (Rule of thumb) [What, Why, How]

GOODLUCK Page No. Date

→ It is technique designed to solve a problem quickly.

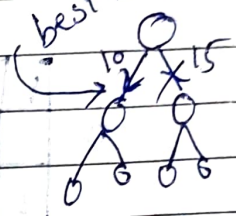
e.g. In 8-puzzle problem, with uninformed search.
there is growth of time complexity
$O(b^d)$ i.e. 3^{20} in worst. - inblind search.

In 15-puzzle & in 24 Puzzle
TC 10^{13} - possible search space. TC - 10^{24}.

i.e. NP - Non-Polynomial Problem.

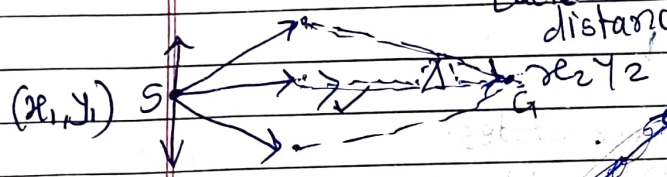
$\boxed{NP \rightarrow P}$

guide or info
best heuristic value



Imp
- It gives good Soln
- but not optimal Soln - no garentee of optimal soln.

Euclidean distance formula $\rightarrow \sqrt{(x_2-x_1)^2+(y_2-y_1)^2}$



formula for quick soln of problem

- Manhatten distance

e.g. 8-puzzle problem with manhatten distance :-

S

1	3	2
6	5	4
	8	7

G

1	2	3
4	5	6
7	8	

for 1 move - $0+1+1+2+0+2+0+2$
$= 8$

OR
find out the misplaced tiles:-
i.e. 5

- It always gives the good soln it is garanteed
- but it does not give the garantee to find the optimal soln.

e.g. Heuristic BFS, DFS, Hill climbing, A* Algorithm

polynomial time. - Heuristic (informed)

* 8-Puzzle problem with Heuristic

[Informed Search]

C BFS,
Greedy algo,
A*, Hill climbing

Possible sol^n = $(100)^3$ = 1 mill

In every min = 5 poss. sol^n Generate
In 1hr = 5×60 = 300 sol^n generate. ~~10 weeks~~
~~In linear search TC = n/2~~
5 lakh sol^{n}s to be generated – 10 weeks required

For prime n
25 25 2
$(25)^3$ = 1 so
only with
two alo
u will g
ur goal
with he

1	2	3
	4	6
7	5	8

S

1	2	3
4	5	6
7	8	

G

Find out heuristic value (H) = no. of misplaced tiles.

h = 3 (dont count space tile □)

0

1	2	3
	4	6
7	5	8

U D R

h=4 h=4 h=2 – minimum value.

1

	2	3
1	4	6
7	5	8

×

1	2	3
7	4	6
	5	8

×

1	2	3
4		6
7	5	8

no need to explore these states.

∴ h=2 is min. value of heuristic search so,

L R U D

h=3 h=3 h=3 h=1 ✓

2

1	2	3
	4	6
7	5	8

×

1	2	3
4		6
7	5	8

×

1		3
4	2	6
7	5	8

×

1	2	3
4	5	6
7		8

2

D = 3

L

1	2	3
4	5	6
7	8	

✓ Goal state

1	2	3
4	5	6
	7	8

3

TC $O(b^d)$ = d=3

SC – $O(b^d)$

for quick sol^n – Heuristic

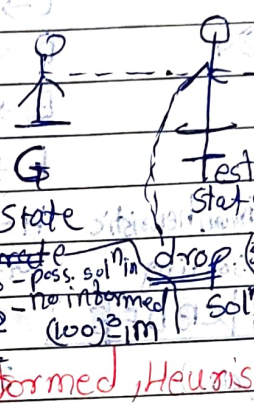
benifit = we convert non-polynomial problem into polynomial problem.
but there is no garantee of optimal sol^n.

* Generate & Test (Heuristic technique, DFS with backtracking)

① Generate a possible solution (continuously) — G → Increase state space
② Test to see if this is a actual sol^n.
③ If sol^n is found, quit otherwise Go to Step 1.

* Properties of Good Generators:-

1) Complete
2) Non Redundant (Redundant states reduced)
3) Informed



Goal sol^n is achieved then accept otherwise go to generate state.

drop (states if sol^n not found)

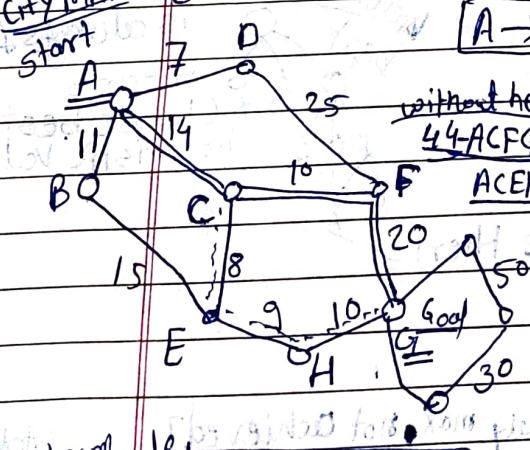
- Two modules — generate & Test
- Let say 1–3 number digit of 2 digits — 100 100 100 — poss. sol^n in no informed — $(100)^3 = 1m$

* Best First Search [BFS] [Informed, Heuristic]

Algorithm:- [Greedy method]

→ Let "OPEN" be a priority queue containing initial state.
Loop
  if OPEN is empty return failure
  Node ← Remove-First (OPEN)
  if Node is a Goal
    then return the path from initial to Node.
  else generate all successors of Node and
  Put the newly generated Node into OPEN
  according to their h values.
  (heuristic values)
END LOOP.

City MAP



$\boxed{A \to G}$ bindout.

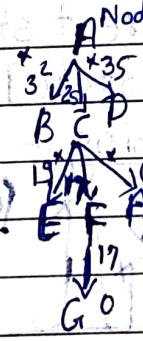
Euclidian distance

straight line distance [Euclidean distance]

without heuristic
44-ACFG ✓
ACEHG-41

$\sqrt{(x_2-x_1)^2+(y_2-y_1)^2}$ — already given

Straight line dist
A → G = 40
B → G = 32
C → G = 25
D → G = 35
E → G = 19
F → G = 17
H → G = 10
G → G = 0



$\boxed{A \mid B \mid D}$
AC $\boxed{B \mid D}$
ACFG ✓

Path from initial to node $\boxed{A \to C \to F \to G}$ ✓

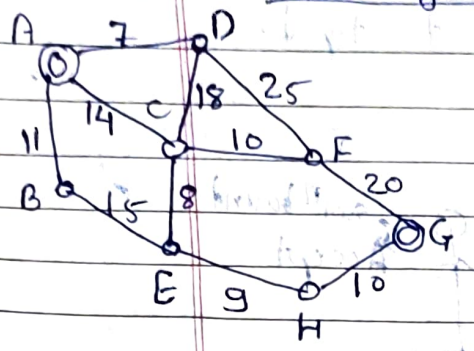
- It gives good sol^n.
- but not optimal sol^n.
- $O(b^d)$ — Time Complexity.

(using Greedy method — min heuristic value & in greedy for that val)
In best case or avg. case informed is better

* Beam Search Algorith [Heuristic]

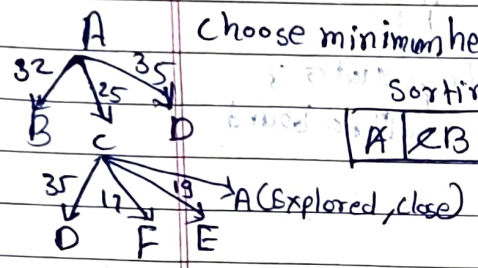
[Slighty change than best first Search]

- Take care of space Complexity (constant).
- Beam width (β) is given.



Straight line distance.

A → G = 40
B → G = 32
C → G = 25
D → G = 35
E → G = 19
F → G = 17
H → G = 10
G → G = 0



Choose minimum heuristic

Sorting

A	C	B	D	F	E

Given value beam:

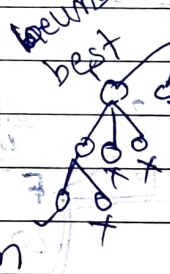
In beam, take β = 2 or n = 2
i.e. take 2 best values

- It is not complete
- It is not optimal.
- It provides quality.
- If β = 1 given, then it is called Hill climbing Algo.

* Simple Hill climbing Algorithm:

[Local search Algo, greedy approach, No backtracking]

1) Evaluate the initial state.
2) Loop Until a solⁿ is found or there are no operators left.
   - select & apply a new operator
   - Evaluate the new state:
     if goal then quit.
   - [if better than current state then it is new current state]



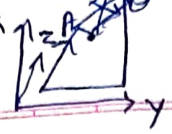
beam width is always 1 means choose best heuristic val.

* Problems in Hill climbing:-

1) Local Maximum: [Globaly max, not achieved]

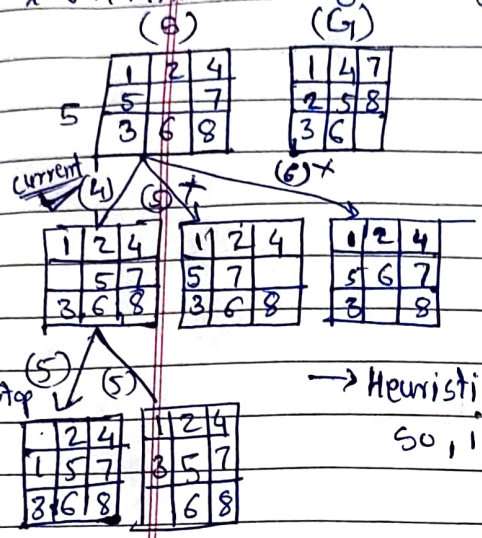
u r blindly climbing the hill & when u feel like fall then u will change ur root & when u get stable then u will stop.

2) Platean / Flat maximum *Heuristic value is same in branc. factor] then it will stop.

3) Ridge  *It cant change direction i.e. in Ridge problem.

* In Hill Climbing u r going with blind & stoping after stable test.

① (blind folded) - Local Search



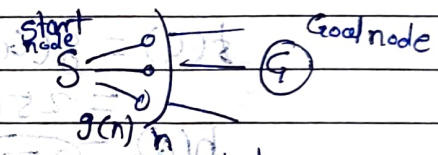
→ Heuristic value increased so, it will stop.

* better space complexity

* A* Algorithm → Informed Searching [Heuristic], Admisible.

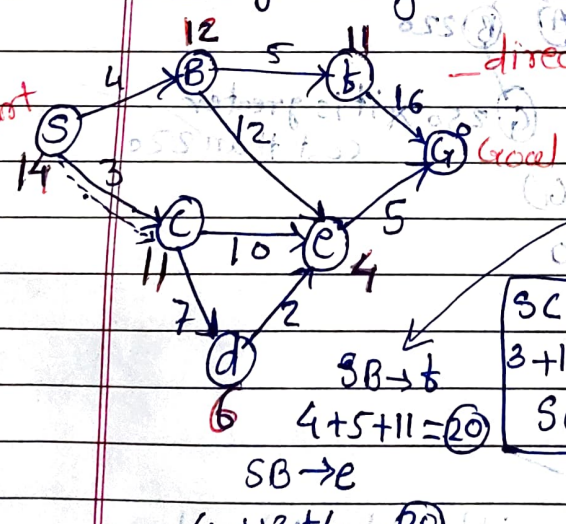
$f(N) = g(N) + h(N)$

(cost) — Actual Cost from start node to n — Estimation Cost from n to Goal node.



A* - * is addmisible. algo.

- It will give u guarantee that optimal solution.



directed

$f(N) = g(N) + h(n)$

$\therefore f(S) = 0 + 14 = 14$

$S \to B$	$S \to C$
$f(S) = 4 + 12 = 16$	$f(S) = 3 + 11 = 14$

$SC \to e$	$SC \to d$
$3 + 10 + 4 = 17$	$3 + 7 + 6 = 16$
$SCd \to e$	$SCd \to e$
~~$16 + 2 + 4$~~ =	$3 + 7 + 2 + 4 = 16$

$SB \to b$: $4 + 5 + 11 = 20$

$SB \to e$: $4 + 12 + 4 = 20$

$SBb \to G = 4 + 5 + 16 + 0 = 25$ ×

$SCde \to G : 3 + 7 + 2 + 5 + 0 = 17$ (G - heuristic by default value)

min. Cost is 17 Goal State achieved.

T. Complexity $= O(V+E)$ & Spac. Comp. $= O(b^d)$.
$= O(b^d)$

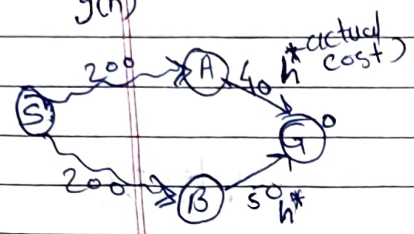
* How to make A* admissible:-

$h(n) \le h^*(n)$ - Underestimation (estimated, Actual)

$h(n) \ge h^*(n)$ - overestimation.

$f(N) = g(n) + h(n)$

$S \xrightarrow{g(n)} n \xrightarrow{\text{heuristic}} G$



$g(A) = 200$

$g(B) = 200$

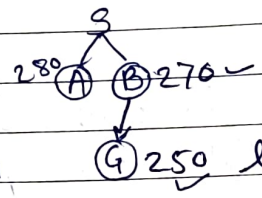
$g(G) = 250$

Case I - Overestimation.

Let say $h(A) = 80$, $h(B) = 70$ $\Big] > h^*$ (i.e. 40, 50)

$f(A) = 200 + 80 = 280$

$f(B) = 200 + 70 = 270$



$f(G) = g(G) + h(G)$

$= 250 + 0$

$h(G) = 250$

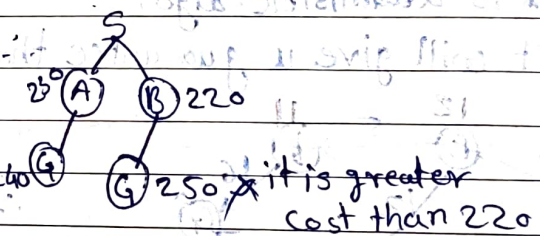
Case-II - Underestimation - optimal sol^n

Let say, $h(A) = 30$, $h(B) = 20$ $\Big] < h^*$

$f(A) = g(A) + h(A)$

$= 200 + 30 = 230$

$f(B) = 200 + 20 = 220$



$\therefore f(G) = g(G) + h(G)$

$= 250 + 0$

$f(G) = 250$

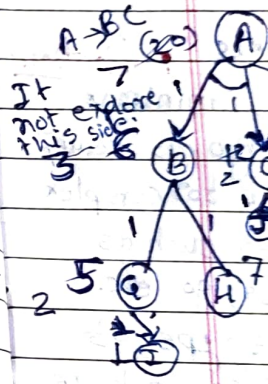
So, $g(G)$ through A is

$g(G) = 240 + 0 = 240$

optimal ans. is done with underestimation

AO* Algorithm (AND OR graph)

e.g.



→ AO* does ... it got ...

* A* & ...

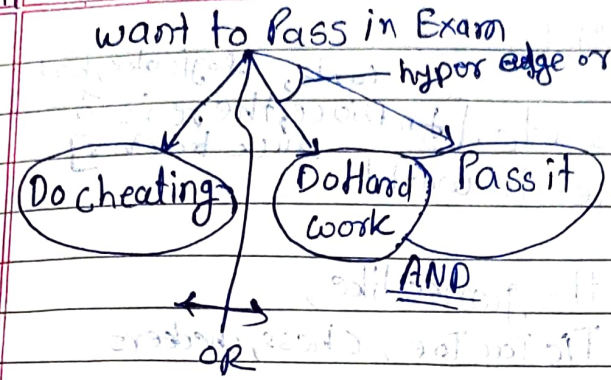
- A* alw... estim...

- But, AO* ...

AO* Algorithm [Problem Decomposition (breakdown into smaller pieces)

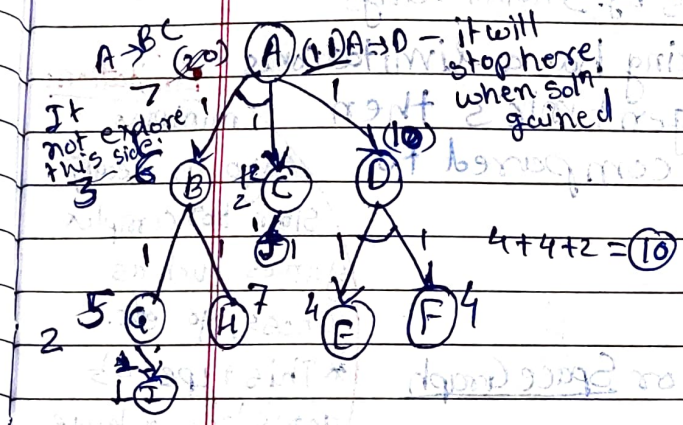
(AND/OR graph) Informed/Heuristic

e.g.

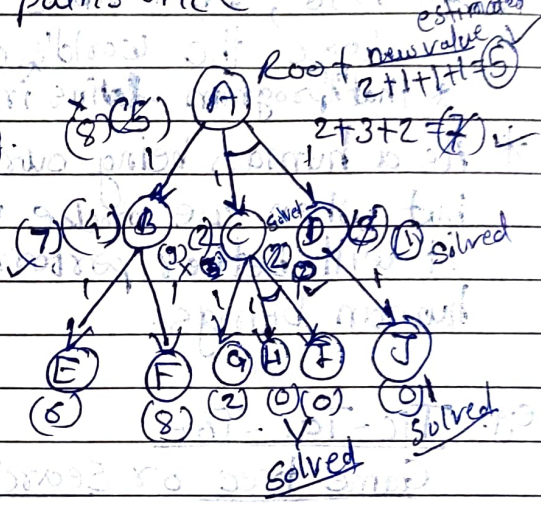


* we know these simple graph, multigraph, or complete graph but now we will see AND OR Graph.

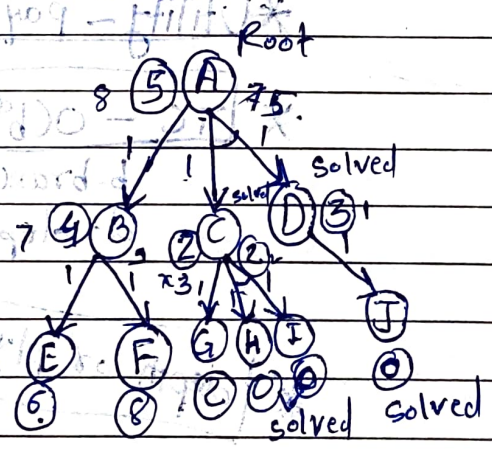
→ AO* does not explore all the sol^n paths once it got a solution.

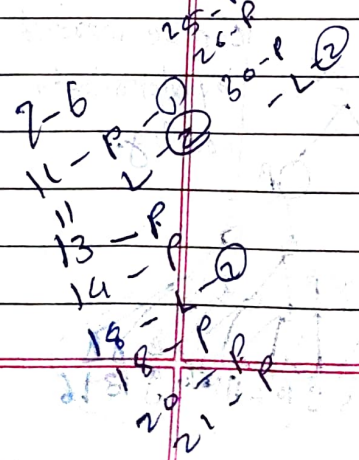


e.g.



* A* & AO* both are informed search techniques:
- A* always gives u optimal sol^n when u works under estimation.
- But, AO* always will not give u optimal solution.





27 = 9 + 18 − P
18 = 6 + 12 − L

* Introduction to Game Playing :-

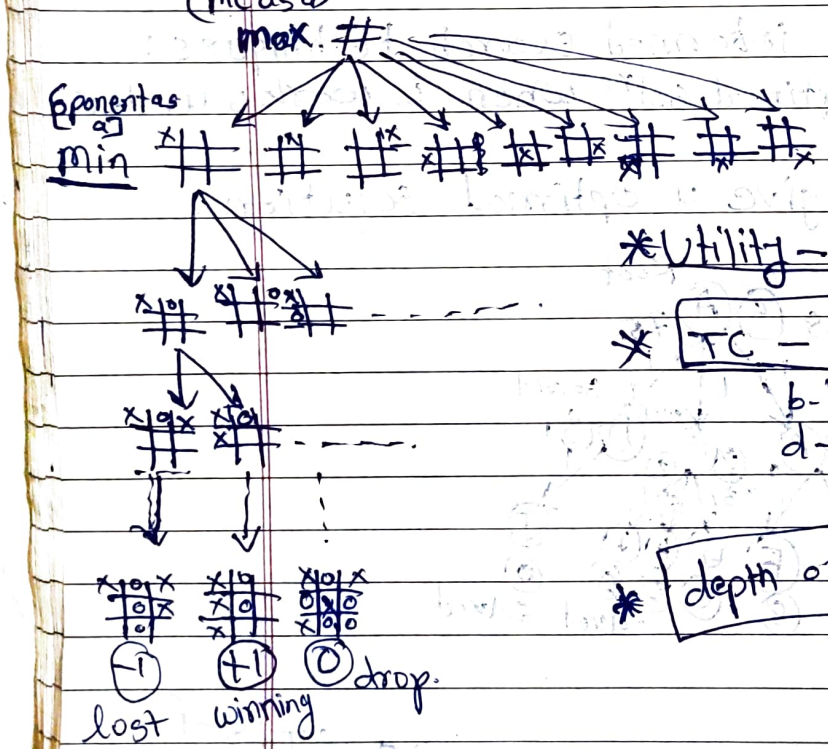
- Minimax Algorithm
- Alpha-Beta (α-β) ~~Pruning~~ Pruning

e.g. PubG, GTA, Call of Duty, Doctors etc. [In Dice, cards, there is a luck factor, logic also, but]
↗ Today's popular games in world.

* In AI, we will talk the games like,
e.g. 8-puzzle ~~problem~~ game, Tic Tac Toe, Chess, checkers

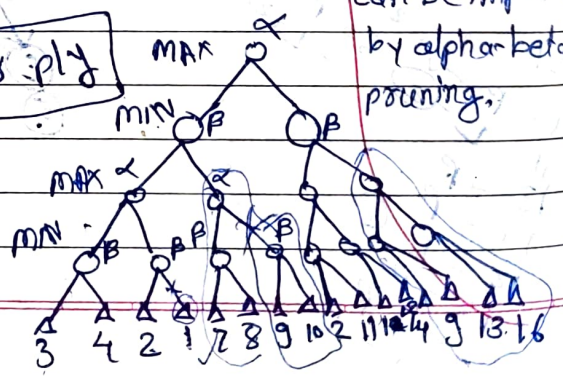
* In 1996, IBM made program, which ~~loose~~ loose the world's chess champian.
* that mlc program ~~wins~~ wins in chess by human beings.
* As a human being our thinking has a limitations but when we make intelligent mlc's then it will better perform as compared to human beings.

e.g. Tic-Tac-Toe
Game tree or Searchtree or Space Graph
(meaning)



* Utility - pay off
* $\boxed{TC - O(b^d)}$
b-branching factor
d-depth.

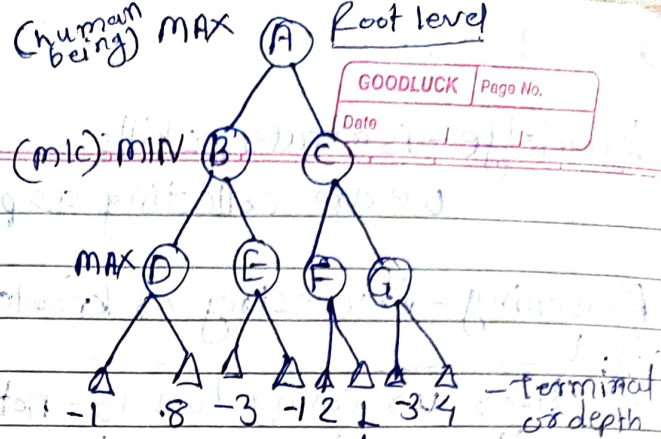
* $\boxed{\text{depth or ply}}$



Limitation
minimax algo. is really slow for complex games such as Chess, go etc.
* This type of game has a huge branching factor (35 choices) for 1 player too 5 moves so, $(35)^{100}$ are possible
& player has a lots of choices to decide.
* This Limitation of minmax algo. can be improved by alpha-beta pruning.

Minimax Algorithm :— DFS

① — Back tracking Algo.
② — Best move strategy used
③ — Max will try to maximize its utility (Best move)
④ — min will try to minimize utility (worst move).



Game tree or Search tree or Space Graph

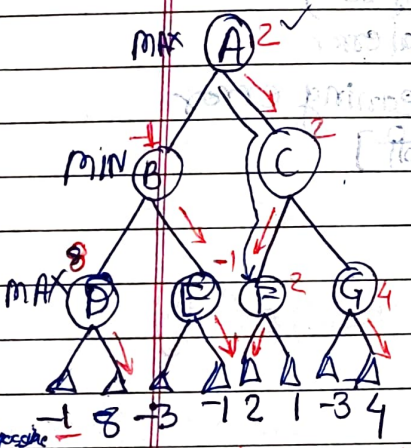
① * BFS will not used in minmax algo. becz, it uses the levelized searching & in minmax one move of max & one move of min i.e. oponent move.

② — max — priority is make best move & win.

② — min — priority is make best move & dont get chance to win for max.

Properties :— 1) Complete : It will definitely find a sol^n (if exist) in the finite search tree.
2) optimal
3) Time Complexity : $O(b^d)$
4) Space Complexity — same to DFS $O(b^d)$

e.g. Tic-Tac-Toe, game



ACF — 2

TC. = $O(b^d)$ where, b-branching factor, d-depth.

e.g. b = 3 d = 2
the $3^2 = 9$ choices.

& In chess moves for 1 play — 50 moves, 2 players — 100 move
$(35)^{100}$ = max. tree, so, we cant use this algo. in some games.
choices

DFS

Limitation: minimax is really complex such as etc. pe of a huge factors ices) very too so, $(35)^{100}$ moves are possible. er has of choices e. Limitat^n max algo. improved harbeta

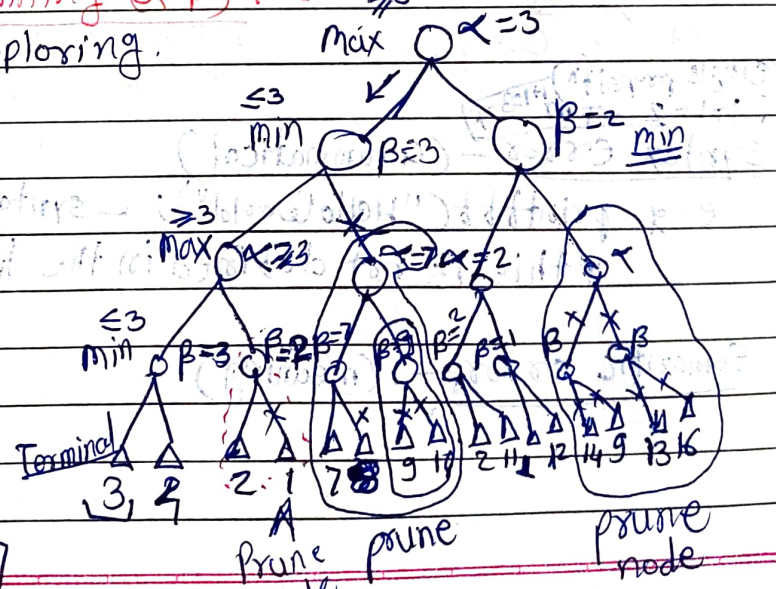
* Alpha-Beta (α-β) Prunning (α-β) :— advanced version of min-max algo.

→ cut off search by exploring less no. of nodes.
— DFS

* TC — $O(b^d)$ — worst case

best case — $O(b^{d/2})$ — TC. higher Complexity.

∴ [α-β prunning is much more efficient than the min-max algorithm] becz, it exploring less nodes & giving less T.C.



* Knowledge Representation & Reasoning :-

GOODLUCK Page No.
Date

Knowledge - is a facts, skills
we are collecting as per our ~~kno~~ educat^n, experience.

Reasoning - Processing of knowledge.

e.g.
On 20-floor u r coming down is - not processing knowledge.
from 20-floor u r not coming down is - Representing knowledge.

→ Logic → Propositional logic - [True & false]
→ Predicate

→ Rules → if then (condition) [m/c acts on if then cond^n]
→ Semantic Net → Google Graph
(meaning graph)
→ Frame → Slots & fillers.
(objects & attributes)
→ Script → (advanced of frames)

when u feed the wrong knowledge to the m/c.
the m/c will Representing wrong output,
errors likes syntax errors = Grammatical error
semantic error = Wrong meaning error
e.g. [मैंने मॅच खाते खाते रोटी देखी]
e.g. I is going to College.

* Propositional Logic :- [Either true or false, not both]

Syntax Semantic

Atomic Complex

(Single propsit^n) 1+1=2 - T or 1+1=3 - F

1+1 = 2 - T
2+1 = 4 - F
New Delhi is Capital of IND - T
but Some studs are intelligent - T/F
& Some studs are not int. - X

Syntax error - (Grammatical)
e.g. printf\b\b("Hello World"); - syntax error
* this is not defined in the library file.

Semantic error - (meaning)

① ¬: Negation : e.g. P: Today is Friday
∴ ¬P: Today is not Friday.

e.g.

T	T	T	T	
T	F	T	F	
F	T	T	F	
F	F	F	F	
		OR	AND	

Truth table OR AND

② ∨ Disjunction: (OR) (You should eat or watch TV at a time)

P: (action) ∨ Q

③ ∧ Conjunction: (AND): (Please like my video & subscribe my channel)

R ∧ M

④ → if then (if there is a Rain then the roads are wet)

⑤ ↔ iff (if & only if): (I will go to the mall iff I hv to do shopping)

Truth Table

T	T	T	T	T	T
T	F	T	F	F	F
F	T	T	F	T	F
F	F	F	F	T	T
		OR	AND	if then	iff
		∨	∧	→	↔

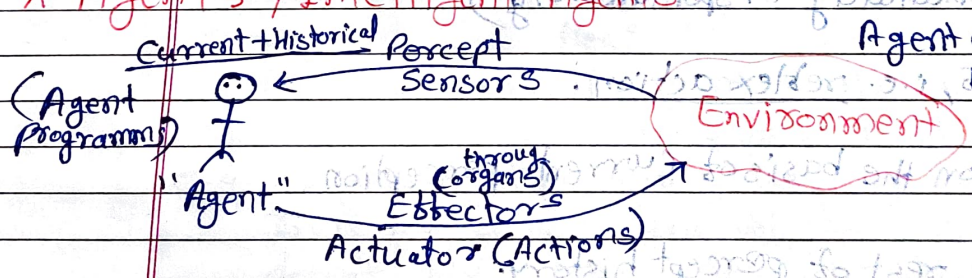
e.g.

* You can access the internet from campus only if you are CSE student or you are not freshman.

P Q R

∴ ~~P ⟹ (Q∨R)~~ ⇒ [P → (Q ∨ ¬R]

* Agents / Intelligent Agents



Agent can sense or percept from env.

Agent → Percept → Decision → Actions → Environments.

e.g. I am going to office & i percept that there will be chance of raining by sensing the ~~big~~ deem light, air flow, humidity, temp through eyes & i percept there raining.

Action → took umbrell & going in env.

Goals of Agents:-

① High Performance

② optimized Result

③ Rational Action:- (Right action)

e.g. ~~Alread~~ already, ~~in market~~ Roborts, Sobia Robort, Dynamic Robort (Google) Kuria Robort, SONY Robort etc. are available in Market.

4 imp. factors of Agents:

P: Performance [Autonomous Cars, Self Decision making, rational action]

E: Environment [Traffic, or Road bikes, or vehicles]

A: Actions: [Break, left-right, Steering, acclerate]

S: Sensors: [Temprature, Rain - wiper moving automatically, through Sensing]

* Types:-

1) Simple Reflex Agents
2) model based Reflex
3) Goal based Agents
4) Utility Based Agents
5) Learning Agents.

① Simple Reflex Agents:-

(Reflex - immediately or spontaneously)

e.g. ~~सी~~ शिंक, i.e. reflex action.

- Acts only on the basis of current perception.

- Ignore the rest of percept history.

- Based on If-Then Rules.

- Environment should be fully observable.

e.g. Tic Tac Toe game.

Knowledge should be to the Agent how to play.

e.g. Chess, (board knowledge)

e.g. Car driving on road (Agent dnt know how many cars or how many pedistrians are there on road)
so, agent is partially observable.

Environment
↓
Percept (sense)
↓
Current situation [what the world is like now]
↓
if then (Condition)
↓
Action

Change

e.g. If Room temp >45°
then switch on AC

[Agent sense the temp. & check condition & performing action]

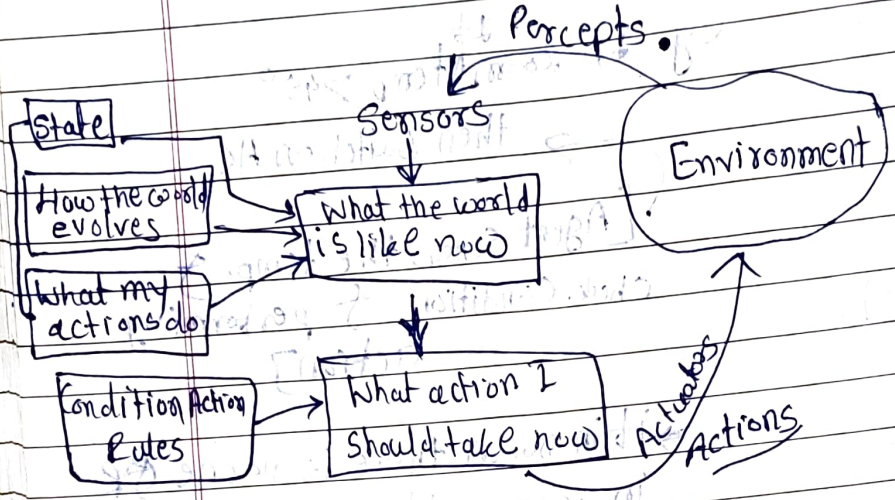
→ if in room there is no one OR in room old age people, they dnt want AC and agent is still performing action, in this situation we want more advanced agents like Model Agents or u should provide multiple conditions like

e.g. if temp >45°
then swith on AC
if room is empty
then Switch on AC.
if there is no old age people in room
then switch on AC.

e.g. If anyone slaps you,
In reblex action you will slap him/her, he doesnt think that who is in front of you.

2) Model Based ~~Learning~~ Reflex Agents :-

Model - (knowledge) based



→ Partially observable Environment

→ Store percept History (Internal Model)

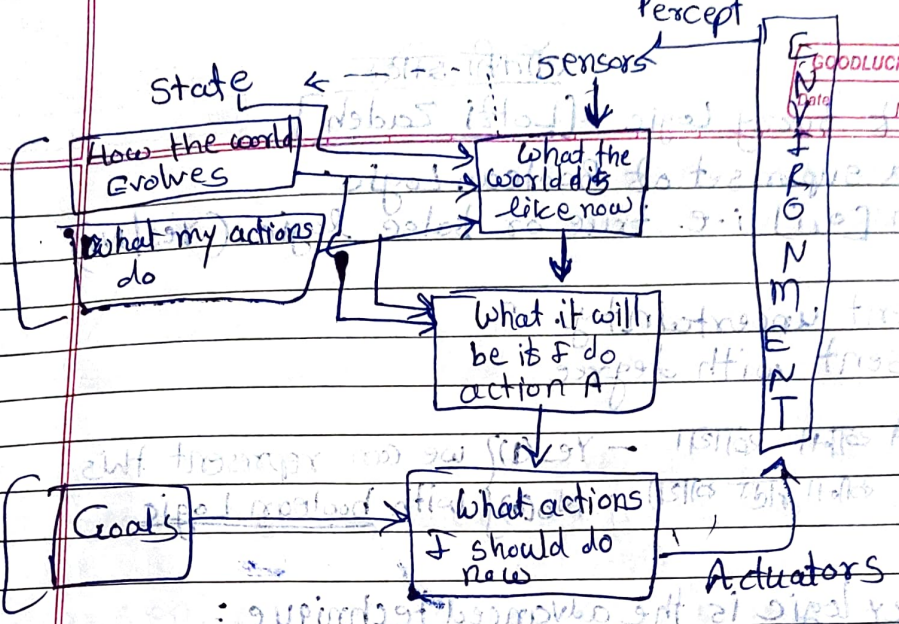
e.g. Vernie (Driverless Cars)
~~[it front car~~

- by providing High level sensors to the cars it is already designed i.e. — Vymu Car [Automated Car]
In USA ↗ not in India.

3] Goal Based Agents - (main focus on Goal)

→ Expansion of Model Based Reflex Agent.
→ Desirable situation (Goal)
→ Searching & Planning

*U know the I/P & O/P i.e. supervised learning
- It is example of supervised learning.

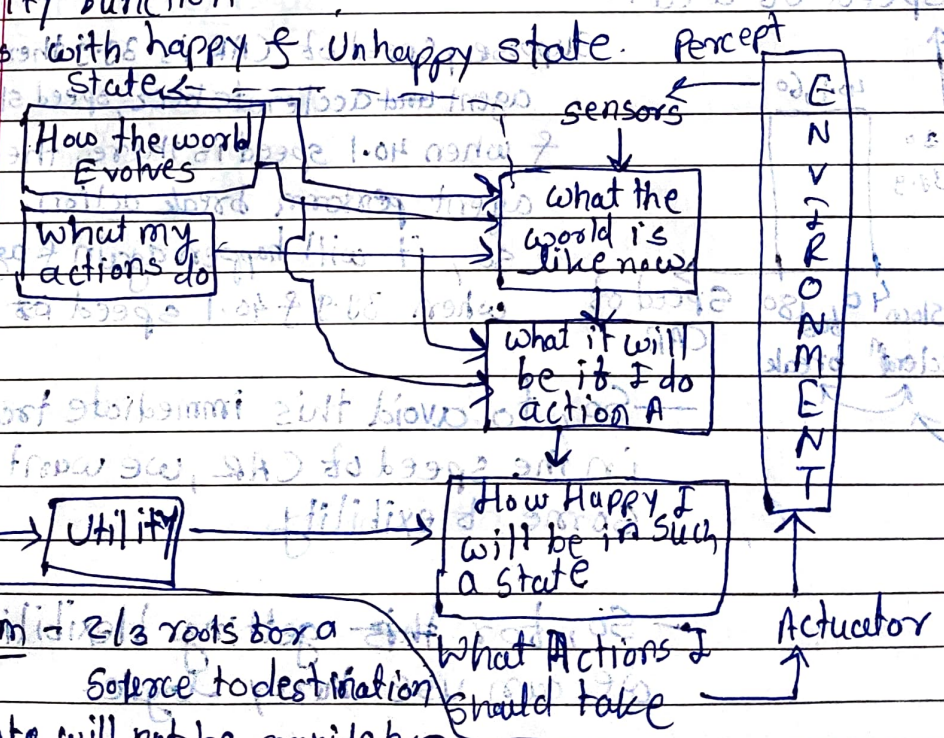


e.g. Amazon — made a lot of Robots
Aibaba (Chinease Company) — Get Robots
↳ delivary of packets at homes

4] Utility Based Agents :— (main focus on utility)

- Partially observable env.
-

1) Focus on Utility not Goal
2) Utility function
3) Deals with happy & Unhappy state.



e.g. GPS sim — 2/3 roots for a source to destination

real life data will not be provide by GPS. So, agent will goes in unhappy state. bcz, on road, there will be chances of accidents, traffic Jam. etc.

in AI

* Fuzzy Logic

लॉतफी ज़ादेह
[Lotfi Zadeh]

GOODLUCK Page No.
Date

- Father of fuzzy Logic - [Lotfi Zadeh]
- It is a super set of Boolean Logic.
  i.e. in [0,1] i.e. true or false logic. (Yes/No)
- Represent uncertainty.
- Represent with degree

e.g. आज मैंने खाना खाया - Yes (1)
आज मैंने खाना नहीं खाया - No (0)
} we can represent this with boolean Logic

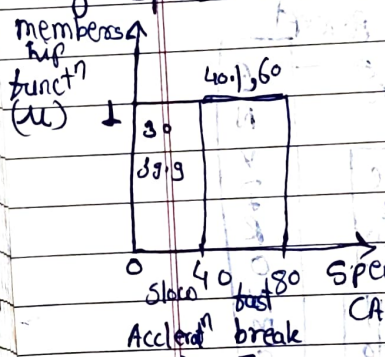
But, fuzzy logic is the advanced technique:

① - Represent uncertainty:
means, we can talk in uncertainty like extended bar.

e.g. मैंने आज बहुत खाना खाया।
मैंने आज कम खाना खाया।
} Here boolean Logic will fail, so, we can use fuzzy Logic.

② Represent with degree:-
e.g. बहुत ज्यादा or बहुत कम can express with degree.

e.g. Speed of a Car.



Suppose, speed of CAR is 39.9 then agent ~~and~~ acclerate becz, speed slow & when 40.1 speed is there, then agent perform break action. So, it will happen again & again when 39.9 & 40.1 speed ~~of~~ car.

झटके मिलेंगे

- So, to avoid this immediate transition in the speed of CAR, we want some flexibility.

- So, for ~~this~~ getting flexibility we can use fuzzy logic.

GOODLUCK Page No.
Date

ogic : (Yes/No)

epresent this
an Logic

ue :

e extended bar.
Logic will fail,
use fuzzy Logic.

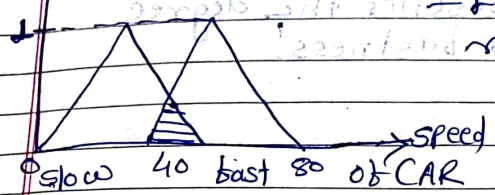
ith degree.

39.9 then
z, speed slow
ere, then
action.
ain & again
eed of car.

te transition
want

bility

membership function (μ)



— It is more accurate way to represent the data.

▲ - In this way we can represent the flexibility means nearest value of 40 is same as 40.
(39.9, 40.1)

→ Represent the belongingness of member of a crisp set to fuzzy set.

eg. Universal set $U = \{1,2,3,4,5\}$
subset $S = \{1,2\}$

we find out, that whether this S belong to U or not?

* According to Boolean : $\{(1,1),(2,1),(3,0),(4,0),(5,0)\}$
x, μ

* According to fuzzy Logic :-

we take through, extension or belongingness.

e.g. 'Check the degree of fastness' of CAR :

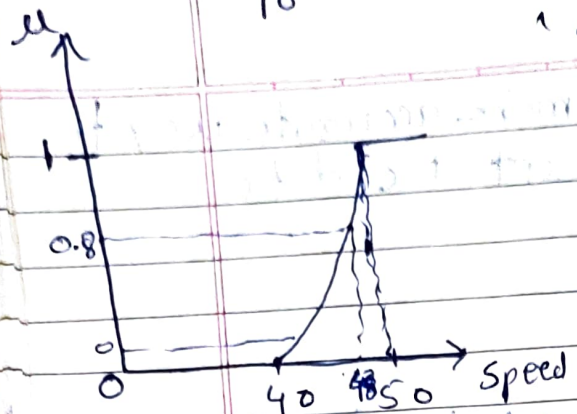
$$\begin{cases} 0, & \text{if speed}(x) \le 40 \\ \dfrac{\text{speed}(x) - 40}{10}, & \text{if } 40 < \text{speed}(x) \le 50 \\ 1, & \text{if speed}(x) \ge 50 \end{cases}$$

Let say speed $x = 30$, $(30,0)$ slow, $x = 60$, $(60,1)$ fast } Boolean logic.

e.g. $x = 42$, $\dfrac{42-40}{10} = \dfrac{2}{10} = \dfrac{1}{5} = 0.2$ } fuzzy Logic

$x = 45$, $\dfrac{45-40}{10} = \dfrac{5}{10} = 0.5$

$x = 48, \quad \frac{48-40}{10} = \frac{8}{10} = 0.8$



'so, fuzzy logic represents the degree of fastness'

* Various operations in fuzzy logic:-

In boolean opn AND, OR, NOT

① Union: $\max\{\mu_A(x), \mu_B(x)\}, x \in U$.
Boolean (OR)

② Intersection: $\min\{\mu_A(x), \mu_B(x)\}, x \in U$
AND

③ Complement: $\mu_{\bar{A}}(x) = [1 - \mu_A(x)], x \in U$
NOT

④ Bold Union: $\mu_{A \oplus B} = \min[1, \mu_A(x) + \mu_B(x)]$

⑤ Bold Intersection: $\mu_{A \odot B}(x) = \max[0; \mu_A(x) + \mu_B(x) - 1]$

⑤ Equality $A = B$ if $\mu_A(x) = \mu_B(x) \; \forall \; x \in S$

Now, $U = \{5, 10, 20, 25, 30, 40\}$

$A = \{(\overset{x}{10}, \overset{\mu}{0.2}), (20, 0.4), (25, 0.7), (30, 0.9), (40, 1)\}$

if $(A = (5, 0.2))$ then u can consider it directly in U but not in n.

$B = \{(10, 0.4), (20, 0.1), (25, 0.9), (30, 0.2), (40, 0.6)\}$

Sometimes $x = \frac{x}{x+2}, \frac{x}{x+5}$ given

∴ ① $\mu_{A \cup B} = \{(10, 0.4), (20, 0.4), (25, 0.9), (30, 0.9), (40, 1)\}$

② $\mu_{A \cap B} = \{(10, 0.2), (20, 0.1), (25, 0.7), (30, 0.2), (40, 0.6)\}$

③ $\mu_{\bar{A}}(x) = \{(10, 0.8), (20, 0.6), (25, 0.3), (30, 0.1), (40, 0)\}$

④ $\mu_{A \oplus B} = A + B = (10, 0.6), (1, 0.6)$ min is $(10, 0.6)$

$\therefore \mu_{A \oplus B} = \{(10, 0.6), (20, 0.5), (25, 1), (30, 1), (40, 1)\}$

⑤ $\mu_{A \ominus B}(x) = (0.2 + 0.4) - 1$

$= 0.6 - 1$

$= -0.4 \quad \max(0, 0.4) = 0.$

$\therefore (10, 0)$

$\therefore \mu_{A \ominus B}(x) = \{(10, 0), (20, 0), (25, 0.6), (30, 1.1), (40, 1.6)\}$

⑥ Equality : suppose $A \neq B$, then inequality.

⑦ Product : –

$\mu_A(x) \cdot \mu_B(x) = (10, 0.08), (20, 0.04)$

*Artificial Neural Networks (ANN) :–

e.g. Google assistant, google translate

Brain ⟶ Machines

'Neural Networks' ⟶ Artificial Neural Networks

e.g. Neuro's are connected to each other. As like in Internet, all PC's, mobile's are connected to each other & sharing of data.

e.g. In human brain, there are millions of nerrows, i.e. very high no. of nurows are connected to each other & it works for ~~pro~~ sharing a data within a very fraction of seconds & generating outputs.

VIMP – i.e. human brain contains learning algo's i.e. from experience it can "train his brain".

* The learning & training capacity of brain is making very imp organ of body.

So, in this way the humans try to give a human brain to m/c's. i.e. Artificial neural network to m/c's.

e.g. Already there are alot of super comp.'s are available & high comput^al devices are available which can do parallel processing, how, these devices are different than ANN ?"

e.g. 8

* within a fracth of sec., m/c's can give answers but humans get more time than m/c's.
becz, human brain is less efficiant than m/c

But, suppose, ur going to office for 1st day, & u meet a lot of people in office & recognize faces of office members.

- And next day, humans brain try to detect or recognize faces, becz, sombody getting shaved, or facial expressions etc. are changed
* even human brain learn & train & detect faces.

* But, same thing in m/c's → ~~we~~ it can scan all the office member's photos on 1st day. & second day, the look changes in office members, the normal computer's cant detect or recognize people.
- means 1st day pics are not 100% matching with 2nd day pics.
becz it's an inaccurate sim.

Normal computers — normal comp.

Input data → Meaning → Learning & Improvement

↓ Algorithens ↓ neural n/w's → [Learning algo's]

ANN

e.g. Avengers (Movie Hollywood) (trending)
Iron Man (Tommy Stark):
- Tonny Stark's pic showing to you & In that no. of diff. ~~typ~~ look's of pics, even u can recognize him., bcz human brain learn & train & recognize, "Yes this is Iron Man or Tonny Stark".

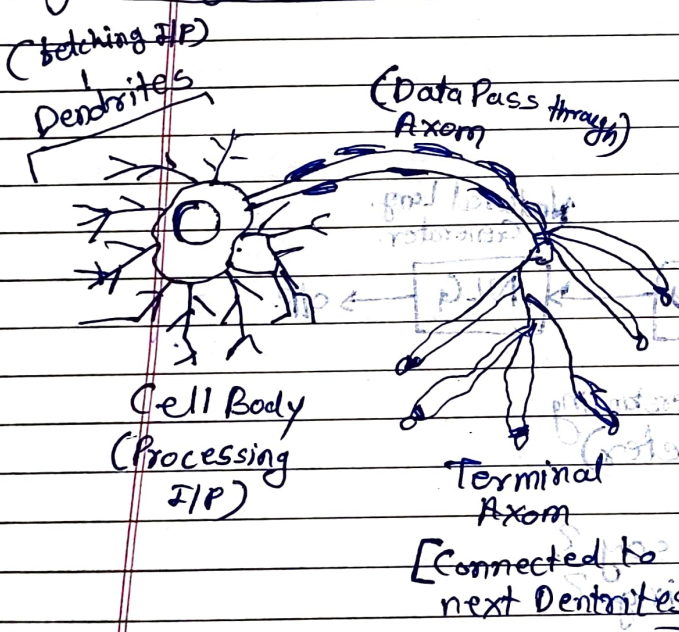
* But, same thing, normal Comp's Cant detect becz, In that ANN is not added.

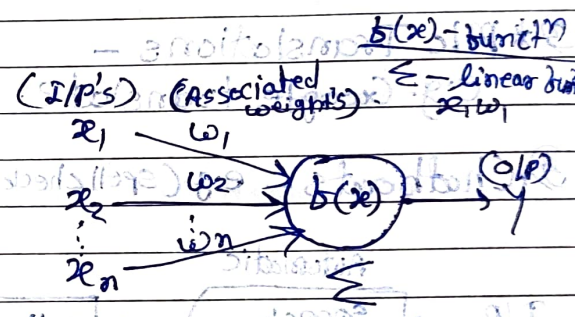
— Bcz, whatever change in pics of that Iron man, u should provide all that into to m/c's.
— On that basis, the m/c's generate the o/p.
— & that o/p will not be much accurate. (It give 40% accuracy)
* — "But if u implement ANN to this m/c's then it will gives 95% to 95% accuracy."

e.g. Advertize of Alia Bhat's :
— i.e. On her friend's T-shirt, there is a Chinese sentence.
— She asked to him, what's the meaning of this?
— He said, sumuri, or philosophy example,
— But, Alia checked it on "Google Translate" then got a exact meaning o/p generated by 'GT'.

* means, Google is already trained with algo.'s & learning so, it will give you meaningfull o/p.

e.g. Biological Neuron's :- (base)





e.g. Google Assistant :- "What is the timing of Avenger's?"
→ It will give you ~~nearest~~ timings & ~~m~~ diff multiplexe's shows available.

means, it leans through the algorithm's & well train & give u accurate o/p.

*Natural Language Processing (NLP) :-

→ How human communicate with each other?

→ Computers should replicate the same thing.

→ Applications of NLP

① * Speech Recognition :-

e.g. Google, Amazon, working on recognition.

e.g. Google Assistant, Siri.

② * Sentiment Analysis :-

e.g. social media.

on facebook, twitter, we can find out the major focus of twits by analysis.

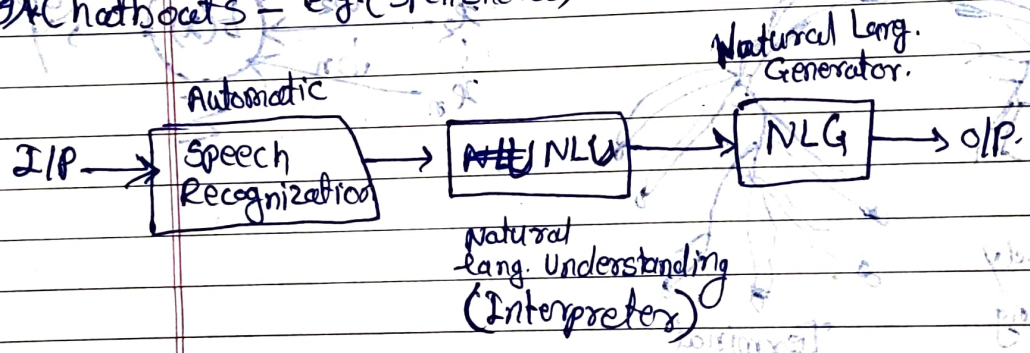
e.g. movies, good or bad - u can analyze

or in politics — major focus analyze.

③ * M/c Translations -

e.g. Google translate.

④ * Chatbots - e.g. (spell checks)



NLU — What do the user say?

their intent? meaning?

Challenges — Lexical Ambiguity:

- Syntactic Ambiguity.
- Semantic Ambigity
- Pragmetic Ambiguity

e.g ① The tank was full of water.
[multiple meanings] Lexical Ambigity.

② old men & women were taken to safe place.
Here, old men & women or old man & old women, this is a syntactic ambigity.

③ The Car hit the pole while it was moving.
⇒ what car or pole is moving, i.e semantic ambigity.

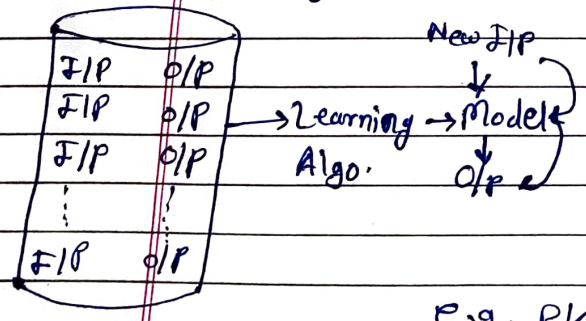
④ The Police are coming.
- multiple meanings, means where coming Police? for U, or me or in this Area?
- i.e. Pregmetic ambigity.

NLG — What should we say to User?
- It should be Intelligent & Conversational.
- Deal with structured data.
- Text / Sentence Planning.

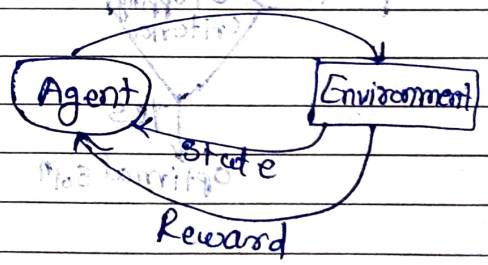
Corpus — knowledge base — Text/sentence plan.

"Types of ML"

* Supervised Learning	Unsupervised Learning	Reinforcement Learning
① Training Data	→ only I/P's	→ Reward/Penalty.
② Both I/P's & O/P's	→ clustering	→ Q-Learning.
③ Classification	→ K-means	
④ Naive Baye's Algo.		



Training Data

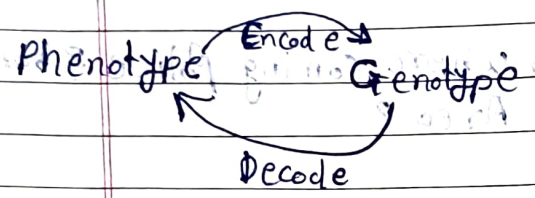


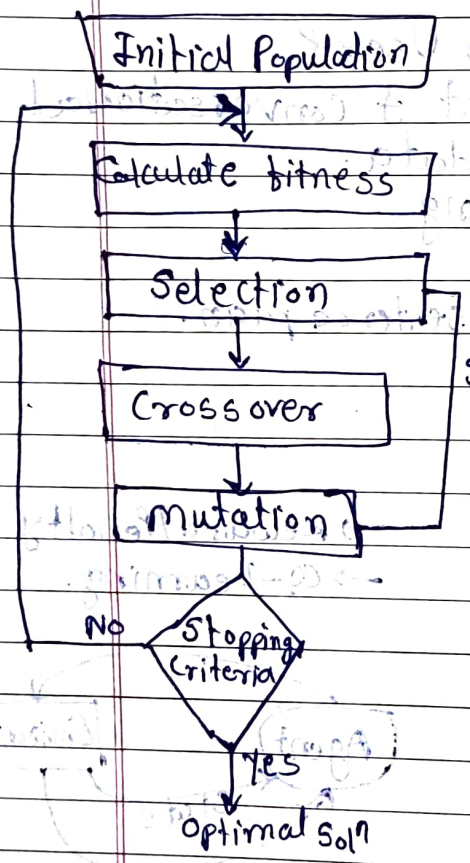
e.g. PK movie, Amirkhan, using Clustering for recognizing people.
i.e. women & men diff.

4. Can implement k-mean, Q-learning algorithm with python or matplotlib.

* Genetic Algorithm in AI :- (John Holland)

- Abstraction of Real Biological Evolution
- Solve complex problems (like NP Head)
- focus on optimization.
- Population of possible solutions for a given problem.
- From a group of individuals, the best will survive.
-





e.g. 4-digit word
I/P - Make
O/P - Take
3 - matching word.
fitness value is 3i
max. fit value, for optimal soln

e.g. abcde
ebghi

abchi } crossover
ebgde }

* Constraint Satisfaction Problem (CSP):-

→ CSP Consists of three components V, D, C.

where, V – is a set of variable : $\{v_1, v_2, v_3, \ldots v_n\}$

D – set of Domains $\{D_1, D_2, \ldots D_n\}$ one for each variable

C – set of constraints that specify allowable combination of values.

$C_1 = (\text{scope}, \text{rel})$

e.g. $v_1 \neq v_2$ $\quad C_1 = ((v_1, v_2), v_1 \neq v_2)$

domain A B $\quad$ or $C_1 = ((v_1, v_2), (A, B))$

Where, Scope – set of variables that participate in constraint.

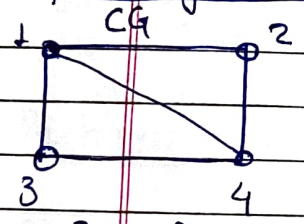
~~Relation~~ rel – Rel is relation that define the values that ~~can~~ variable can take.

Let say v_1, v_2 $\quad \therefore C_1 = (v_1, v_2)\ (1,2)(1,4), (2,4)$

(1,2) (2,4)

How Constraint Satisfaction Problem (CSP):-

– Intelligent backtracking (DFS).



$V = \{1, 2, 3, 4\}$

$D = \{\text{Red, Green, Blue}\}$

$C = \{1 \neq 2, 1 \neq 3, 1 \neq 4, 2 \neq 4, 3 \neq 4\}$

vertex

	1	2	3	4	
Initial Dom	R,G,B	RGB	RGB	RGB	
1 = R	R	GB	GB	GB	
2 = G	R	G	GB	B	
3 = B	R	G	B	B	Empty.

[problem not solved, so int. backtrack]

	1	2	3	4
∴ 3 = G	R	G	G	B

problem solved.

Artificial Intelligence - AI

(Syllabus)

AI - Theory - 3hrs/week
Tut - 1hr/week

Theory. ESE - 70 marks
TW - 25 marks
CIE - 30 marks

Pre-requisite: Basic Pgm.ing in Python.

① Artificial Intelligence & Its Issues:-
Def's, - Importance of AI Evolution of AI - Applications of AI, Classification of AI s/m's w.r.t. environment knowledge Inferring s/ms & planning, Uncertainty & towards Learning s/m's.

② Overview to Problem Solving & Heuristic Search:-
Problem solving by search, Problem space - state space Blind search - Types, Performance measurement. Types, Game playing mini-max algorithm, Alpha-Beta prunning.

③ Probablistic Reasoning & Markov Decision process:
probability, conditional probability, Bayes Rule, Bayesian Networks - Representation, construction & inference, temporal model, hidden Markov model. MDP formulation, utility theory, utility functions, value iteration, policy iteration & partially observable MDPs.

④ ~~Reinforcement~~ Learning s/m's & Expert Systems:-
Forms of Learning types:- Supervised, Unsupervised Reinforcement Learning, Learning Decision Trees. Expert Systems - Stages in the development of a ES - Probability based Expert s/m's - ES Tools - Difficulties in Developing ES's - Applications of ES's

⑤ Reinforcement Learning:-
Passive reinforcement learning, direct utility estimati

adaptive dynam
reinforcement l

⑥ AI with python
study of impor
NumPy, Scip
Python. Setti

Study of rea
Case studies
swift AI,

① Write a pgm
search.

② Write a pgm.

③ Write a pgr
from given

④ Write a pgm

⑤ Write a pgr
grid world

⑥ Write a pgr
world.

⑦ Develop a s
i> Predic
ii> Detec
iii> Ident
iv> Shock

⑧ Case Stud

Theory: ESE - 70 marks
TW - 25 marks.
CIE - 30 marks

n of AI - Applications
w.r.t. environment
ng, Uncertainty &

tic Search:-
pace - state space,
asurement.
rithm, Alpha-Beta

ion process:
Bayes Rule,
nstruction &
Markov model.
functions,
rtially

Systems:-
, Unsupervised,
Trees.
t of a ES -
ls -
ons of ES's.

bility estimation

adaptive dynamic pgm.ing, temporal diff. learning, active reinforcement learning - Q-learning.

6) AI with Python:-
study of important inbuilt libraries of python like NumPy, Scipy, matplotlib, nltk, simpleAI. Installing Python. Setting up PATH. Running Python.

Study of real time applications of AI with python, Case studies: AI Platforms - Azure ML, Google AI swift AI, Tensorflow.

Tutorial List - 8

1. Write a pgm. to conduct uninformed & informed search.
2. Write a pgm. to conduct game search
3. Write a pgm. to construct a Bayesian network from given data.
4. Write a pgm. to infer from the Bayesian n/w.
5. Write a pgm. to run value & policy iteration in a grid world.
6. Write a pgm. to do reinforcement learning in a grid world.
7. Develop a small AI based mini project like:
   i) Predicting user's next location.
   ii) Detecting YouTube comment spam
   iii) Identifying the genre of a song.
   iv) Shock front classification.
8. Case Study on any one real time AI application.

Artificial Intelligence (AI)

① History of AI :-

Talos : was a giant animated bronze warrior programmed to guard the island of Crete Created by Hephaestus

* these are the ideas but nobody knows the actual implementation.

- ML & AI were thought of a long ago.

- 1950 - Intro. of AI.
  - Alen Turing Published a paper, in which he ~~schedut~~ speculated about the possibility of creating machines, that think like human beings.
  - So he was created the Turing test. This test is basically used to determine whether ~~a~~ or not a comp. can think intelligently like human being.
  - He noted that thinking is difficult to define. but the meachine will think that is meaning that the m/c will pass the turing test

*Still todays date We haven't found that the bully cleared m/c

1951 - This was also known as the game ~~of~~ AI
  - So, in 1951, by using Ferranti Mark 1 m/c a comp. scientist known as Christopher strachey wrote the checker's pgm. & at the same time a pgm was written for chess as well.

Now, these pgm.'s are well & redone.

or that would complete with humans in playing chess

1956 — probably the most imp year in the invention of AI.
— Bcz, first time, the AI was coined.

— So, the term AI was coined By John McCarthy at the Darthmouth Conference in 1956.

1959 — The first AI Laboratory was established. This period marked the research era for AI.

MIT Lab which is still running til date.

1960 — In 1960, the First Robot was introduced to the general Motors assembly line.

1961 — the First chatbot was invented
e.g. Siri, Alexa — Now
but, in 1961 — Eliza Chatbot.
this is followed by famous IBM Deep Blue.

1997 — The news broke down, that the IBM's Deep Blue beats the world champion Garry kasparov, in the game of chess.

— So this is kind of first accomplishment of AI.
— It was able to beat the world champion at chess.

2005 :— When the DARPA Grand challenge was held, A Robotic car named Stanley, which was built by Stanford's racing team. won the DARPA Grand Challenge.
— There was a another big accomplish of AI.

2011 - IBM's question answering system, Watson, defeated the two greatest Jeopardy Champ, Brad Rutter & Ken Jennings.

* So, guyzz this is how AI evolved.

- Starts from hypothetical situation, right now it's the most imp technology in todays ~~world~~ world.

Vimp
* - If we look around in our daily ~~world~~ life, everything around us is run through AI & deep learning or ML.

- So since emergence of AI in 1950's - exponential growth & its potential.

So, AI covers domains such as ML, deep learning & neural networks, natural language processing, knowledge based, expert S/m's & so on. Comp. vision & image processing.

* Question:- AI comes in half of century, but suddenely it get's important. why? are we talking about AI now?

Main reasons for the demand of AI:-

① We hv more computation power now
So, AI requires ~~mor~~ alot of Computing power.
- Recently, many advances hv been made & Complex deep learning models are deployed.
- & one of the greatest technology that made this possible are GPUs.

② We have a lot of data at present.
- we are generating data at an immeasurable pace
e.g. through Social media, - insta, facebook, whtsapp etc
through IOT devices.

* So, we need to find out method or solution, that can help us process this much data & help us derive useful insight.
- So, we can grow business with help of data.

e.g. when u shop online, recommed the data for you & how to classify an object from an image.

③ Better Algorithms -
ⓐ Neural networks - (Concept behind the deep learning)
It is done with more accuracy
so, the demand of AI is increased.

④ Universities, governments, startup, & tech giants are all investing in AI.
e.g. Google, Amazon, Facebook, Microsoft.
have heavily invested in AI

- bcz, they believe that AI is the future.

JIMP
* So, AI is rapidly growing both as a field of study & also as an economy.

What is AI?

AI first coined by
- 1956 - John McCarthy

- AI as a science intelligence of making m/c's intelligent
- is the theory & development of comp. s/m's able to perform tasks that normally require human intelligence.
  e.g. visual perception, speech recognition, decision making & translation between languages.

Imp
* AI is a technique to getting m/c's which work & behave like humans.

e.g. Health care, robotics, marketting, business analytics, & many more.

Applications of AI :-

① Google predicting search :-
- when u begin typing a search term & google makes recommendations for you to choose form
- i.e. nothing but AI in action.
- So, predictive searches are based on data that google collects about you, such as your browser history, your location, your age & other personal details.
- So by using AI, Google attempts to guess what you might be trying to find.

Behind this, there's alot of NLP - Natural Lang. Proc, deep learning & m/c learning involved.

- but the logic behind the google search engine is AI.

② In finance Sector :-

J.P. Morgan — contract intelligence platform using ML & AI & image recognition s/w to analyze legal documents.

* AI's computational power is very strong as compared to humans.
bcz, it will gives quick reply within a fraction of seconds.

③ Healthcare -

IBM Watson — is one of the pioneers that has developed AI s/w specifically for medicine.

* more than 230 Healthcare organizations use IBM AI technology.

2016 — IBM Watson technology was able to cross referen 20 million oncology records quickly & currently diagnose a rare leukemia condition in a patient.

The Google's AI eye doctor —
- where they were working with an Indian eye care chain to develope AI s/m.
- which can examine retinal scan & identify a conditional called diabetic retinopathy which can cause blindness.

④ Social Media :-

1) Facebok — AI is used for face verification
- where you make use of ML. & Deep Learning Concepts. in order to detect facial features & tag your friends.
- Auto tagging features, behind that ML, DL & NN

Instagram, facebook, Twitter — AI

- Twitter's AI -

* To identify any sort of hate speech & terroristic languages in tweets.

- again it uses ML, DL & NLP in order to filter out any offensive or any reportable content.

Mom
I AM sad

* Virtual Assistants :-

e.g. Siri & Alexa

newly released -
Google's virtual assistant - Google Duplex.
astonished - millions of people around the world
- not only can it respond to calls & book appointments for you.
& it also adds a human touch.
So, it adds human fillers & all of that.

- It is hard to distinguish betn human & the AI speaking over the phone.

* Self driving Cars :-

- AI implements image detectn, DL, in order to build cars. that can automatically detect any object or any obstacles & drive around without human intervention.
- So, these r fully automated self-driving cars.

Elon musk - talks alot about how AI is implemented in test class driving cars.

Tesla Company - fully self-driving car's.

robo taxi version - that can ferry passengers without anyone behind the wheel

AI is taken by tech giant companies - like Google, Tesla, Facebook - all these data driven

Netflix :- with AI & ML, Netflix has developed a personalized movie recommendations for each of its users.

- If you open Netflix & look at type of movies that are recommand to you, they are diff.
- Bcz, Netflix studies each users personal details, & tries to understand what each user is interested in & what sort of movie patterns each user has & then it recommends movies to them.

- So Netflix uses the watching history of other users with simill. taste to recommend.

- So, there recommendation engine is more powerfull.

Gmail :- Gmail has seperate section, i.e. Spam mails.
- Concepts of AI & ML Algorithms to classify emails as spam & non-spam.

Spam mails - lottery, earn, bull emails
by using - by using ML & NLP & AI.

Types of AI :-

1] Artificial Narrow Intelligence — Weak AI

2] Artificial General Intelligence

3] Artificial super Intelligence.

* ANI - is only exists right now

1] ANI — Weak AI

e.g ① Alexa — example of weak AI
— It operates within unlimited predefined range of functions.
— there is no self awareness

e.g. ② The Google Search engine, Sophia the humanoid, self-driving cars, famous Alphago
Comes under weak AI

— So guyzz, right now we are in the stage of ANI or weak AI

— We actually hvn't reached ~~AGI~~ AGI or ASI
but lets look at wht exactly it would be like

2] AGI — Strong AI

It makes m/c's to perform any intelligent task that a human being can
— m/c dont posses like human ability.
— m/c's hv very strong processing unit, that can perform high level computations,

— If you tell m/c to ~~m~~ walk up in your room & switch on TV,
m/c will take forever to learn that
— bcz, machine dnt hv resonable way of thinkinking.

— they hv not capable of thinking & reasoning like human being
— So far, we hvnt developed any m/c that can fully be called strong AI.

e.g. AlphaGo Zero which defeated AlphaGo in game of Go
— basically learned in span of 4 months
— but even it not classified as a fully strong AI.
bcz it cant reason like a human being.

3] ASI — will surpass of a human being.

— hypothetical situation as depicted in movies & any science fiction books
wherein m/c's hv taken over the world

e.g. Terminator movie — that depict ASI.
These dont exist yet

* Some people's speculate that ASI will take over the world by the year 2040.

* any m/c are thinking or reasoning like huma beings is not possible yet now,
so AGI & ASI is still in process.

* Programming Languages for AI :

* N no. of Lang.s

1) Python — is probably most famous language. for AI
- most effective lang. for AI.

- Syntaxes belongs to python are very simple & they can be learned very esily.
- One of the most easiest lang. is python.
- & AI & ML algorithms can be easily implemented in Python.
- Bcz, there are a lot of liabrariies, which are predefined functn's for these algo.s

* 2) R Lang :-
3) Java
4) LISP — John McCarthy — father of AI
5) Prolog — knowledge base & ES
6) C++
7) JavaScript
8) Scala
9) Matlab
10) Julia

Python has all the possible pkg.s,
① It is very easy to understand & learn

* Python - 1989

Features of Python } → - very easy pgm.ing lang.
= very easy to understand
- very easy to grasp this lang.

① Interpreted, object-oriented Lang.
② High level pgm.ing Lang.
③ It also free & open source
④ Platform independant (Portable)
e.g. Linux, windows, Macintosh, Solaris & so on.

⑤ diff. paradigm — object-oriented & procedure oriented pgming

⑥ It is extensible means, it can invoke C & C++ liabrariies

⑦ Python is actually gaining unbelievable huge momentum in AI.

⑧ The Lang. is used Data science algo. & ML algo.s & IOT projects

⑨ less code required, bcz, there are predefined packages, that hv all the functn's & algo. stored.

* Liabrariies in Python

⑩ Pibrain — used for ML.
NumPy — scientific computation
Pandas — & so on

* Introduction To Machine Learning :-

- The ML is used in AI
- The ML is a method through which you can feed a lot of data to a m/c & make it learn
- AI is a vast of field.

AI
- (ML) machine learning
- (NLP) Natural Lang. Processing
- (ES) Expert Sim
- image recognition
- object detection
- (DL) Deep learning

defn
- AI is a methodology in which you make m/c's mimic the behaviour of human beings.

- ML is a way of feed a data to the m/c, so that they can make its own decision.

* Need for Machine Learning

- Why ML ?

- need for ML begins since the technical revolution itself.

- As we generate 2.5 quintillion bytes of data every single data every single day.
- & it is estimated that by this year 2020, 1.7 mb of data will be created every second for every person on earth.

- I am generating a lot of data.

- Now You are watching the vedio on this Youtube, also accounts for data generation, so there's data everywhere.
- To find useful insights & deliver more accurate results.

e.g. Netflix, Amazon – Build such ML models by using tons of data

- Now we, hv a lot of data, so, we can find a way to analyze, process & draw useful insights from this data.
- in order to help us grow buisnesses or to find solⁿ's to some problems
- Data is the solⁿ.
  we just need to know how to handle the data.

help- ML, DL & AI – to handle data.

* ML is important –

① Due to increase in data generation, so, due to excessive production of data, we need to find the method to structure, analyze & draw useful insights from data.

this is why ML comes in

② To improve decession making, by making use of various algo.s, ML make better buisness decessions

e.g. ML is used to predict any drawn balls in the stock market.

① identify any sort of risk & anomalies.

③ Uncover patterns & trends in data.
- building predictive models & statistical techniques.
- But if you use ML algo.s, you can perform computatⁿs in less than a second.

④ Solve Complex Problems –
- from detecting genes linked to deadly ALS disease
- to building self-driving cars.

- Star's which are 2,400 light years away from our planet

* So, ML is very important at present, bcz, we r facing a lot problems with data, we r generating a lot of data. & we r to handle this data.

* What is Machine Learning ?

- Arthur Samuel – in year 1959 which 3 years after AI was coined.

- most of technologies based on ML today.
- most of AI technologies are based on ML technology & DL.

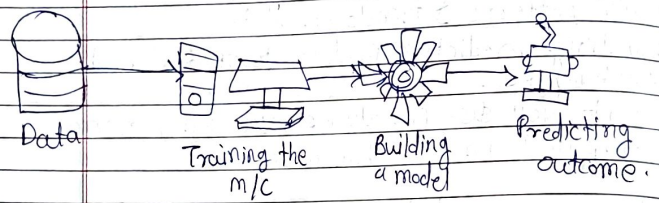
Defⁿ:– A comp. pgm. is said to learn from experience E w.r.t. some class of task T & performance measure P if its performence at task in T, as measured by P, improves with experience E.

② AI in ML is subset of AI which provides machine experience ability to learn automatically & improve

Date

*How can m/c's think or make decesions?

- If you feed a m/c a good amount of data, it will learn how to interpret, process & analyze data by using ML algo.s



- It involves data & training the m/c & build the model for predicting outcome or to find a soln to problem (by using m/c learning algo.)

* Machine Learning Definitions -

1) Algorithm - Set of rules & statistical techniques used to learn patterns from data.
e.g. - linear regression or decesion making tree or random forest.

2) Machine Learning Model -
- A model is trained by using a m/c Learning algo.s
- Model is a main component of ML
- an algo. maps all the decesions that a model is supposed to take based on the given i/p in order to correct o/p

- So, the model will use ML algo, in order to draw useful insights from the i/p & give u an o/p i.e very precise

3) Predictive variable.
- used the data that can be used to predict the o/p

Date

eg. You r trying to predict the height of a person depending on his weight.

so, here your predictive variable is weight.

4) Response variable:-
It is the feature or o/p var. that needs to be predicted by using the predictor variable (s).

e.g. now, in above example - height - response variable.

- target var. of o/p variable.

5) Training Data - Train the model
splitting the data into 2 parts - is data slicing -

1 part is - training data & 2 part is - testing data.

- The ML model is built using the training data.

6) Testing Data - test the model
After the model is ~~tested~~ built, it must be tested in order to evaluate how accurately it can predict an outcome.

*ML Process

- building the predicting model that can be used to find the sol^n for a problm.

steps -
1. Define object of our problem.
2. Data gathering
3. Preparing ur data
4. Data Exploration
5. Building a Model.
6. Model evolution
7. Predictions.

Problem -
1. U need to predict the occurance of rain in your local area by using ML?
→ 1) what r we trying to predict.

- The o/p is continuous or discrete var.

2) What are the target features?
3) What is I/P data.
^ temprature, or humidity level so on..
4) What kind of problm are we facing?
Binary classificat^n? clustering? regression problem.

Step-2 - Data Gathering.
- such as weather cond^{n}s, humidity level, temprature, pressure, etc. collected manually or from web.

Step-3 - preparing Data.
4 - Exploratory Data analysis
5 - Building a mlc learning Model - ML algo.s = linear regression & Decision making.

Step-6:- Model Evolut^n & optimization.
7 - Predictions = whether rain or not it is catogarical variable

*Types of Machine Learning:-

1] Supervised Learning:-
It is basically a technique in which, we teach or train the m/c by using the data which is well labeled

e.g.

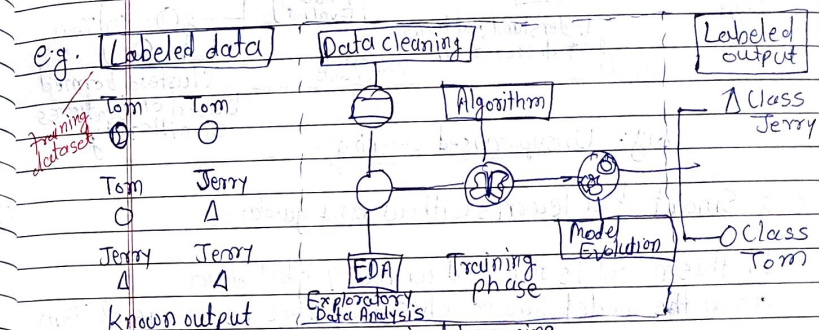


fig: Supervised Learning.

e.g. A kids ~~nee~~ have a lot of problem while solving ~~pr~~ math problem.
So, our teachers always help kids to understand acted^n n how it is done?

i.e. The labeled data set is a teacher, that will train you to understand the patterns in data.
- So, the labeled data set is the training data set.

fig: Here we feeding the images of Tom & Jerry to m/c & the Goal is for m/c to identify & classify into two classes.
- The training data set is labeled.
- basically, we are telling m/c's, this is how Tom looks & this is how jerry looks.
- By doing this we r training the m/c by label data

2] Unsupervised Learning –

– Training the m/c by using ~~unsupervised~~ unlabeled data.

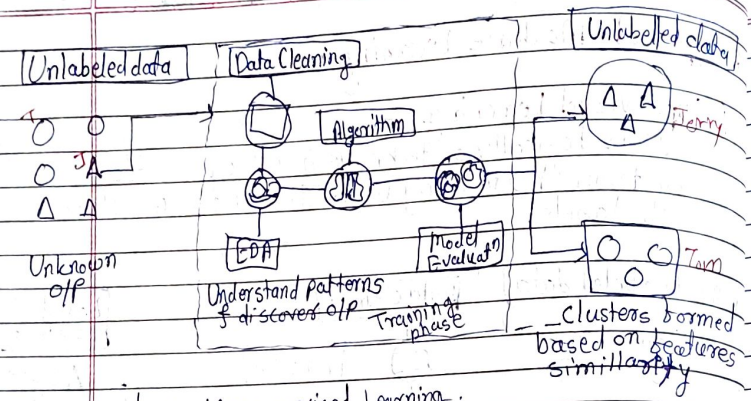


fig: Unsupervised Learning.

e.g. Smart kid learns without any guidance

– In this model is not fed with any label data as in the model has no clue that this is image of Tom & this is jerry

– This is bcz of pattern & differences betn Tom & Jerry on its own by taking in tons & tons of data.

– e.g. It identifies prominent features of Tom: such as pointy ears, bigger in size, & so on...
– To understand that this image is of type-1.

– Similarly, it finds features of Jerry – ~~such as~~ & knows that this image is of ~~Jerry~~ type-2

– It forms two diff clusters.
It forms cluster-1 – which r very similar.
other cluster-2 – is very diff from cluster-1

– that's how unsupervised learning works.

– The m/c has to understand the o/p on its own. bcz, it has unlabled data

– Finally the m/c forms clusters based on ~~feature~~ similarly.

3] Reinforcement learning –

– quite diff from previous 2 types of ML

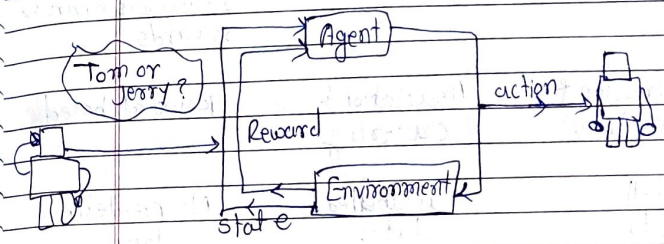


fig: Reinforcement Learning.

– Where, an agent is put in an environment & he learns to behave in this environment by performing certain actions & observing the rewards which is gets from those actions.

e.g. imagine u would droped out in isolated island. Now panic. ofcourse initially well panic. but as time passes by, you will learn how to live on island.

– You will explode the environment, u will understand the climate conditions, the dangers of island, the food on island so on....

– So, this is exactly how reinforcement learning works.

e.g. AlphaGo – reinforcment & Deep learning

– It's like a newborn baby with no information about anything.

11] Difference

	Supervised Learning	Unsupervise Learning	Reinforcement Learning
① Definition	The m/c learns by using labelled data	The m/c is trained on unlabelled data without any guidance	An agent interacts with its environment by producing actions & discovers errors or rewards.
② Types of problems	Regression & Classification	Association & Clustering	Reward based
③ Type of Data	Labelled Data	unlabelled data	No pre-defined data.
④ Training	External supervision	No supervision	No supervision
⑤ Approach	Map labelled I/P to known o/P	Understand patterns & discover o/P	Follow trial & error method.
⑥ Popular algorithms	Linear regression, Logistic regression, Support vector m/c, KNN, etc.	K-means, C-means etc.	Q-Learning, SARSA etc.

	Regression	Classification	Clustering
①	Supervised L	• Supervised L	• Unsupervised L
②	o/P is a continuous quantity. (e.g. weight)	• o/P is Categorical quantity. (e.g. emails categorised into spam or non spam) (& gender of person etc)	• Assigns data points into clusters.
③	Main aim is to forcast or predict.	• main aim is to compute the category of data.	• main aim is to group of similar items clusters.
④	e.g. Predict stock market price	• e.g. classify emails as spam or non-spam.	• e.g Find all transact^ns which are frauduleni in nature
⑤	Algo: Linear Regression	Algo: Logistic regression	• Algo: K-means

e.g. Netflix - clusters there users into similar groups, based on their interest based on their age, geography & so on...

— by using K-means unsupervised Learning.

e.g. To predicts the house pricing index is the Linear Regression problem, bcz price is continuous quantity

e.g. Whether to approve loan of an applicant based on his profile.

→ KNN Algorithm. (Classification)

Approve Loan. Reject Loan

12] Types of Problem Solving using ML:-

- classified in to 3 types i.e.

③ e.g. K-means Algo.

Popular movies — Non-popular movies.

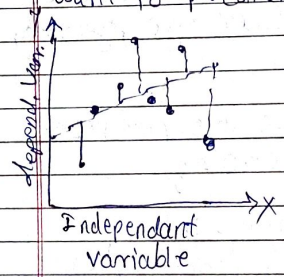
13) Supervised Learning Algorithms.

1. Linear Regression
2. Logistic Regression
3. Decision Tree
4. Random Forest
5. Naive Bayes classifier
6. K nearest neighbour.
7. Support vector m/c

1) Linear Regression :- Algorithm.

Is a method to predict dependant variable(Y) based on values of independant variables (X)

It can be used for the cases where we want to predict some continuous quantity.



$$Y = \beta_0 + \beta_1 X + \epsilon$$

e.g. To predict the price of stock over a period of time

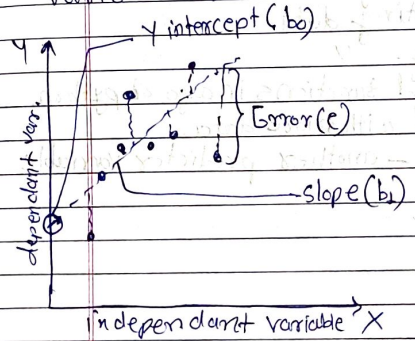
- For such a problem u can make use of linear regression, by studying the relnship betn dependant variable = stock price. & independant variable = time

- So, u are going to predict the how ~~vari~~ whereas the price of stock over a period.

- So, price of stock is - dependent var or o/p variable
  Time is — predictor variable or independant var.

- stock price is continuous quantity.

- Draw out a relnship betn dependant & independant variable. in the best fitting linear line.



$$Y = \beta_0 + \beta_1 X + \epsilon$$

Error, independant variable, Slope, Y intercept, dependant variable

X — min. temp.
Y — max. temp.

Linear Regression Demo

& min. temperature on a particular date

```
import seaborn as SeabornInstance.
from sklearn.model_selection import train_test_split
from sklearn.lear_model import LinearRegression.
from sklearn import metrics

# %matplotlib inline.

dataset = pd.read_csv ('C://Users//sahiti//Desktop//
                AI Full Course Dataset//weather.csv').

print (dataset.shape)
print (dataset.describe())
```

[// we assign max. dataset for training the m/c
//So, that it can predict the data outcome.
// 80% = For training data
20% = for testing data.
In data splicing. //
// predict is predefined functions in algo. of python.
// Training your model with more data.
// Parameter tunning – another predictor variable.

2] Logistic Regression Algorithm: (Classification algo.)

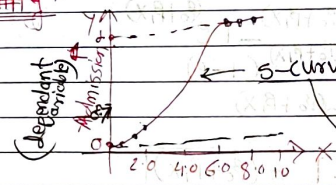
e.g. Your Little sister trying to grad in another school & u want to predict whether she'll get admitted in her dream school or not?

- ok, so based on her CGPA & the past data, we use logistic regression to force the outcome.
- to predict a categorical outcome.

* Linear regression is used to predict the continous quantity variable, rather than a categorical variable one, which is used in Logistic Regression algo.
* whether it is zero or one, it is categorical. So, Linear regression not hv this ability.

def^n – Logistic regression is a method to predict a dependant variable, given a set of independant variables such that dependent variable is categorical - outcome quantity.

Predict^n – [1 or 0, True or false or Yes or no]



[0.7 - probability value means ur outcome is 1!]
(Sigmoid curve)

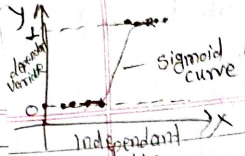
How it works?

– Linear Regression equat^n = $Y = \beta_0 + \beta_1 X + \epsilon$.

dependent var. → Y
Y intercept → β_0
Slope → β_1
independant variable (used to predict Y) → X
error → ϵ

– Represents a relationship bet^n $P(X) = Pr(Y=1|X)$ and X?
(Probability of Y ~~gir~~ = 1 or given some value of X)

- It categorize the probable values in 0 or 1.
- The values must be positive & it lies in bet^n of 1
it must be 1 < value or 0 > value.



① • Take the exponent of the eqⁿ, since the exponential of any value is a positive number.

② • Secondly, a number divided by itself + 1 will always be less than 1

i.e. $P(x) = \frac{e^{(\beta_0+\beta_1 X)}}{e^{(\beta_0+\beta_1 X)}+1}$

$P(e^{(\beta_0+\beta_1 X)}+1) = e^{(\beta_0+\beta_1 X)}$

$P \cdot e^{(\beta_0+\beta_1 X)} + P = e^{(\beta_0+\beta_1 X)}$

$P = e^{(\beta_0+\beta_1 X)} - P \cdot e^{(\beta_0+\beta_1 X)}$

$P = e^{(\beta_0+\beta_1 X)}(1-P)$

$\frac{P}{(1-P)} = e^{(\beta_0+\beta_1 X)}$

$\boxed{\ln\left[\frac{P}{(1-P)}\right] = (\beta_0+\beta_1 X)}$

- we calculate value in betⁿ 0 & 1 i.e probability. also we categorize the value in 0 or 1 i.e. classification
- So, o/p - can be probability ranging betⁿ 0 & 1. thats why it shows a sigmoid curve.

3] Decesion Tree :- Algorithm: (supervised ML algo.) (Classification algo.)

- each node represent a predictor variable, the line betⁿ the nodes represents a decesion & each leaf node represents an outcome (response var.)

e.g. You hosted a huge party & you want to know how many of your guests were non-veg eaters.

decesion var. — Did the guest eat chicken? (best attribute) — Root Node

Branch

Yes → Non-Veg

No → Did the guest eat mutton? — Internal node

Yes → Non veg

No → Did the guest eat sea-food?

Yes → Non-veg

No → veg — Leaf or Terminal Node.

Root Node — The root node is the starting point of a Tree, At this point, the first split is performed

Internal Nodes — represents a decesion point (predictor variable) that eventually leads to the predⁿ of a outcome.

Leaf/Terminal Nodes — represents the final class of the outcome & ∴ they were also called terminal nodes.

Branch — Connection betⁿ the nodes, responses - Yes or No (0 or 1)

[ID3 Algorithm — Iterative Dichotomiser 3 Algorithm] one of most effective algo. which is used to build decesion tree.

Concepts used - entropy & information gain

ID3 algo. 6 steps :-

Step: 1) Select best Attribut (A) - (Predictive variable)

Step: 2) Assign A as a decision variable for the root node.

Step: 3) For each value of A, build a descendant of the node

Step: 4) Assign classificatⁿ lable to the leaf node (i.e. veg & Non-veg)

Step: 5): If data is correctly classified, then you stop at that.

Step: 6): Else : iterate over the tree.

Information Gain & Entropy

Problem:- To study the data set & create a decision tree that classifies the speed of a car as either slow or fast.

(o/p variable)

Road type	obstruction	Speed limit	Speed.
Steep	Yes	yes	Slow
Steep	no	yes	slow
flat	yes	no	fast
Steep	no	no	fast

Step-1

① How do you know which var best seperates the data?

→ The var. with the highest inform^n gain best divides the data in to the desired o/p classes.

· Calculate the foll. measures:

1] Entropy

2] Inform^n gain (IG)

Entropy :- measures the impurity or uncertainty present in data.

IG — indicates how much "information" a perticular feature/variable gives us a significant measure.

Calculating IG of parent node (speed of car)

· P(slow) → fract^n of 'slow' outcomes in Parent node

· P(fast) → fract^n of 'fast' outcomes in Parent node.

Formula: P(slow) = no. of slow outcomes in parent node / total no. of outcomes.

$P_{slow} = \frac{2}{4} = 0.5$

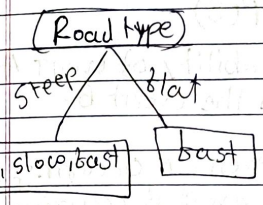
Simi^y, $P_{fast} = \frac{2}{4} = 0.5$

$\therefore$ Entropy(parent) $= -\{P_{slow} \log_2$

Entropy parent $= -\{0.5 \log_2(0.5) + 0.5 \log_2(0.5)\}$

$= -\{-0.5 + (-0.5)\}$

$= 1$



Entropy of LHS

$P(slow) = 2/3 = 0.667$

$P(fast) = 1/3 = 0.333$

$\therefore$ Entropy (LHS child node) $= -\{0.667 \log_2$

$= -\{-0.38 + (-0.52)\}$

$= 0.9$

4] Random Forest Algorithm :-

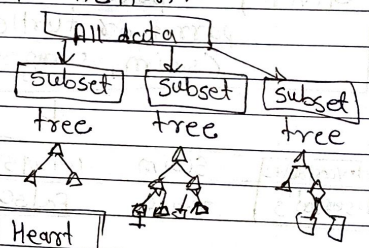
- builds multiple decision trees (called the forest) and glues them together to get a more accurate & stable prediction.

Why Random Forest?

① More accuracy

② Avoid overfitting.

③ Bagging



e.g. root node root node

Blood flow	Blocked Arteries	Chest pain	weight	Heart desease
Abnormal	No	No	130	No
Normal	Yes	Yes	195	Yes
Normal	No	Yes	218	No
Abnormal				

~~Sell — 513.50~~

~~Buy. — 513.85~~

5) NAIVE BAYES Algorithm

- Is based on Bayes Theorem i.e. used to solve classif problems by following a probabilistic approach.

- It is based on the idea that the predictor variabl in a ML model are independant of each other.

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

Where, $P(A|B)$: Conditional probability of event A occuring, given the event B.

$P(A)$: Probability of event A occuring.
$P(B)$: Probability of event B occuring.
$P(B|A)$: Conditional probability of event B occuring, given the event A.

e.g.

o/p variable (Type)	Predictor variables: Swim	Wings	Green	Sharp teeth
Cat	450/500	0	0	500/500
Parrot	50/500	500/500	400/500	0
Turtle	500/500	0	100/500	50/500

Given, To predict whether the animal is a cat, parrot or turtle based on defined pred. variabl (swim, wings, green, sharp teeth).

↓

Pred. variables	Swim	Wings	Green	Sharp Teeth
Observat^n	True	False	True	False

To solve this use, Naive Bayes Theorem:-

$\Rightarrow P(H|\text{multiple Evidences}) = P(C_1|H) * P(C_2|H) \ldots * P(C_n|H) * P(H) / P(\text{multiple evidences})$

Class : I.4(CC) Semest

ect : Ja...

esti

* To Check if the animal is a Cat:

$P(Cat \mid swim, Green) = P(swim \mid Cat) * P(Green \mid Cat) * P(Cat) / P(swim, Green)$

$= 0.9 * 0 * 0.333 / P(swim, Green) = 0$

value is '0' so, it is not Cat.

⇒ For Parrot :- ?

$P(Parrot \mid swim, Green) = P(swim \mid Parrot) * P(Green \mid Parrot) / P(swim, Green)$

$= 0.1 * 0.80 * 0.333 / P(swim, Green)$

$= 0.0264 / P(swim, Green)$

⇒ Simillarly, For turtle.

$P(Turtle \mid swim, Green) = P(swim \mid Turtle) * P(Green \mid turtle) / P(swim, green)$

$= 1 * 0.2 * 0.333 / P(swim, green)$

$= 0.0666 / P(swim, green)$.

probability of turtle is > Parrot.

∴ The animal is actually a turtle ✓

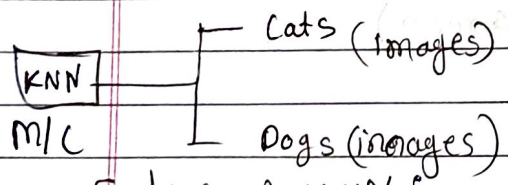
- This is how Naves Bayes works.
- basically you calculate the condnal probability.

6) K-Nearest Neighbour (KNN) Algorithm

K-Nearest Neighbour is a supervised Learning algo. that classifies a new data point into the target class depending on the features of its neighbouring data points.

- If we want a m/c which distinguish betn images of cats & dogs.
- we should train a model with certain features.

- features such as pointy ears used to identify cats.
  long ears for dogs.



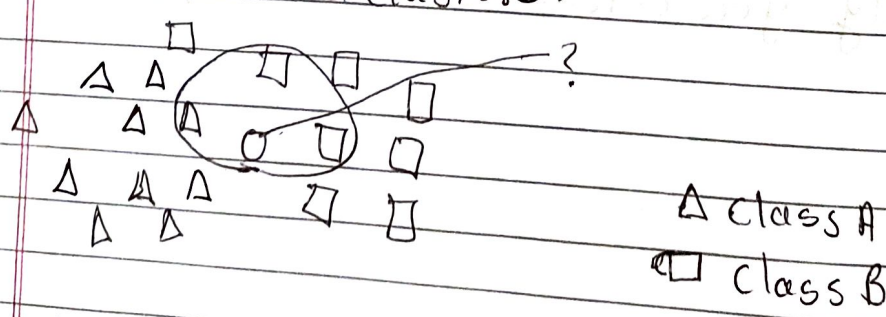
Features of KNN :-

- Supervised learning algo.
- It is labeled dataset.
- It is simple
- It is Non-parametric

e.g. Naive Bayes is a parametric model.

- It is Lazy algo.
- It has classificatn & regression algo
- Similarity features.

* Problem stmnt: - Assign a new data point into one of the 2 clusters.



- K - stands for no. of nearest neighbours

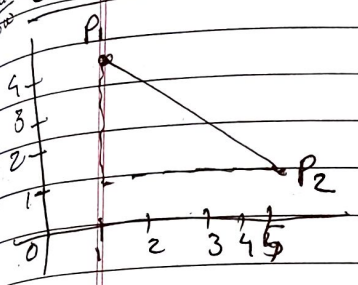
Consider, value of K as - 3

bcz, 3 neighbours are closest to the new data point O

calculated by – Euclidean distance, & Manhattan dist.
– so $k=3$ – 2-squares □ & 1-Δ – then data point O is assigned to class B
or suppose $k=7$, i.e 3=□, & 4-Δ
majority of class A, so it O is assigned to class A

e.g. we can use elbow method.

– Euclidean Distance :–



= measure dist. betn P_1 & P_2.

Point $P_1 = (1,1)$

$P_2 = (5,4)$

$= \sqrt{(x_2-x_1)^2+(y_2-y_1)^2}$

Eucledian dist. $= \sqrt{(5-1)^2+(4-1)^2} = 5$

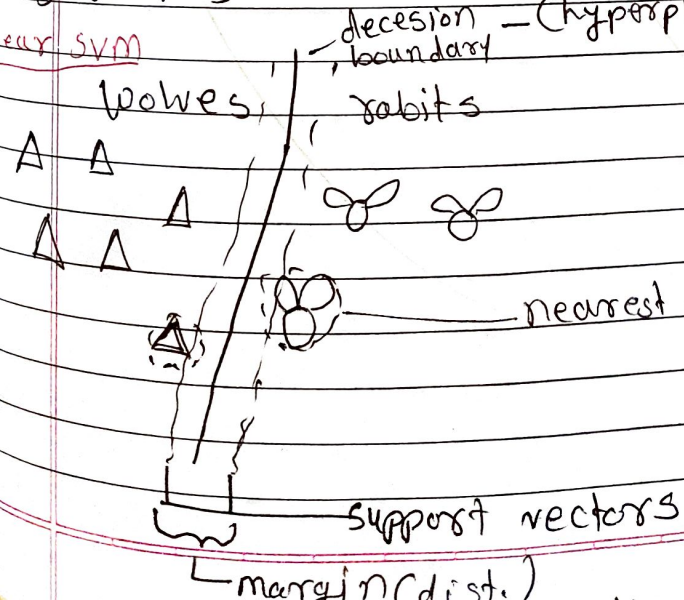
So, the dist. of Point P_1 & $P_2 = 5$.

7) Support Vector Machine (SVM) :–

- Its a classification Algorithm.
- SVM makes use of something known as a hyperplane – decesion boundary betn seperate classes
- It uses regression & Classificetn Algo.
- Support vector regressor – for processing.
- Also classify non-linear data using kernel tricks

eg. protect the farmer his rabits from ~~wolfs~~ wolves.
& he needs to build the fense.
– he needs to draw a decesion boundary betn rabbits & wolves

Linear SVM



– this is for linear data structure.

AI - Unit - 3

* Interming systems & planning :-

3) Probabilistic Reasoning in AI

* Uncertainty :-

- With knowledge representation we might right
* A → B, which means
  If A is true then B is true
but when we dnt know that A is true or not then we cant express this stmnts, this situation is called uncertainty.

- So to represent uncertain knowledge, where we are not sure about predictions we need uncertain reasoning o probabilistic reasoning.

* Causes of uncertainty :-

1. Informatn occured from unreliable sources.
2. Experimental Errors
3. Equipment fault.
4. Temprature variatn
5. Climate change.

* Need of Probabilistic Reasoning in AI:-

① When there are unpredictable outcomes

② When specificatn or possibilities of predicates becomes too large to handle.

③ When an unknown error occurs during an experiment.

* In probabilistic reasoning there are 2 way's to solve problems with uncertain knowledge:

① Bayes' rule
② Bayesian networks.

* Probability :- Can be defined as a chance that an uncertain event will occur.

- The value of probability always remains betn 0 & 1 that represents ideal uncertainties

① $0 \leq P(A) \leq 1$, where $P(A)$ is probability of an event A.

② $P(A) = 0$, indicates total uncertainty in an event A

③ $P(A) = 1$, indicates total certainty in an event A

Formula:

$$\text{probability of occurance} = \frac{\text{No. of desired outcomes}}{\text{Total no. of outcomes}}$$

- $P(\neg A)$ = probability of a not happening event
- $P(\neg A) + P(A) = 1$

① Event :- Each possible outcome of a variable is called an event.

② Sample Space - The collectn of all possible events

③ Random Variables - used to represent the events & objects in real world.

④ Prior probability :- probability computed before observing new informn.

⑤ Posterior Probability :- The prob. i.e. calculated after all evidence or informatn has taken into account.

* Conditional Probability :-

- Is a probability of occuring an event when another event has already happened.

e.g. Let's suppose, we want to calculate event A when event B has already occured, "the probability of A under the conditions of B",

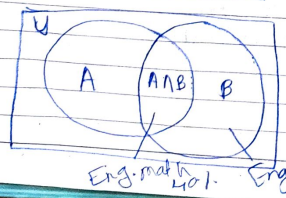
- written as:

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

where, $P(A \cap B)$ = joint probability of A & B.
$P(B)$ = Marginal prob. of B.

* If prob. of A is given & we need to calcul. prob. of B then

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



Venn diagram

e.g. In a class 70% of students who likes English & 40% of Students who likes Eng. & math. & then what is the % of students those who like English also like maths?

Soln :-

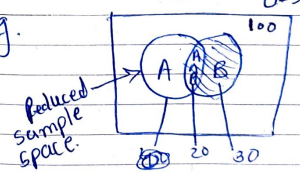
A is an event that studs like Engl.

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{0.4}{0.7} = 57\%$$

∴ 57% are studs, who likes Eng. also like maths.

* If u want to understand Bayes theorem the u must know that what is Conditional probability.

e.g.



$$P(B) = \frac{30}{100} = 0.3$$

$$P(A \cap B) = \frac{20}{100} = 0.2$$

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{0.2}{0.3} = 0.67$$

* Bayes Theorem :-

$$P(A/B) = \frac{P(B/A) \cdot P(A)}{P(B)}$$

Posterior — Likelihood — Prior — Marginal.

* Conditional Probability :-

(II) $P(A|B) = \frac{P(A \cap B)}{P(B)} \Rightarrow P(A|B) \cdot P(B) = P(A \cap B)$ — RHS

given that event B has already occurred

(I) $P(B|A) = \frac{P(B \cap B)}{P(A)} \Rightarrow P(B|A) \cdot P(A) = P(A \cap B)$ — LHS ~~R.H.S~~ RHS

so, from (I) & (II)

$\therefore P(A \cap B) = P(A|B) \cdot P(B) = P(B/A) \cdot P(A)$

$$\Rightarrow \boxed{P(A/B) = \frac{P(B|A) \cdot P(A)}{P(B)}}$$

A - is hypothesis
B - evidence (data)

VIMP
* We have to find the probability of hypothesis given that we have been given a data OR we have observed some evidence

[5-minutes Engineering YTube]

VIMP
* We cant directly compute posterior, for that we should take the help of other probabilities

③ Prior Probability :- is the probability of hypothesis before observing or before considering the evidence

② Likelihood Probability - of the evidence given that the hypothesis is true.
- u should find out evidence or data probability on the basis of hyposis is true

④ Marginal :- Pure evidence of probability given data or evidence pure probability

e.g. 52 - cards in game.

$$\boxed{P(King|Face) = \frac{P(Face/King) \cdot P(King)}{P(Face)}}$$

$= \frac{1 \cdot 4/52}{12/52}$

Queen - 4
king - 4
jack - 4

$= \frac{1 \cdot 1/13}{3/13} = \frac{1}{3}$

* Applications of AI :-

① It is used to calculate next step of robot when the already executed step is given.

② is helpful in weather forecasting

③ It can solve the monty Hall problem.

* Bayesian Belief Network in AI :-

* Used to solve a problem which has uncertainty.

- A Bayesian n/w is a probabilistic graphical model which represents a set of variables & their conditional dependancies using a directed acyclic graph.

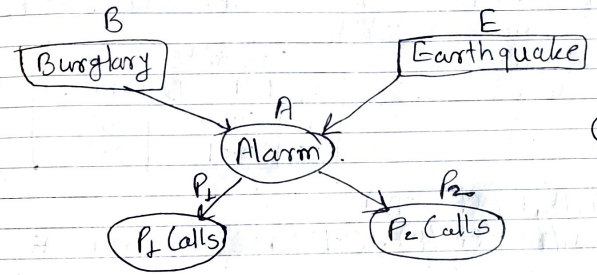
- It is also known as:
  Bayes network OR
  belief n/w OR
  decision n/w OR Bayesian model.

- Bayesian n/w graph is made up of nodes & arcs (directed links).

- The Bayesian n/w graph doesn't contain any cyclic graph.
  Hence, it is known as directed Acyclic Graph (DAG)

Bayesian n/w's are probabilistic bcz, these n/w's are built from a probability distribution.
& also

① Directed acyclic Graph :- DAG



S home owner

- P_1 - Sometime the P_1 will not call while Alarm rings bcz his ringtone & Alarm's ringtone is same.

- P_2 - P_2 can't be a good listeners when he is too far from the alarm so thats why he missed out the ringing alarm.

Problem : Calculate the probability that alarm has sounded, but there is neither a burglary, nor an earthquake occured, & P_1 & P_2 both called the S

$P(B=T) = 0.001$ $\quad$ $P(E=T) = 0.002$

$P(B=F) = 0.999$ $\quad$ $P(E=F) = 0.998$

observed values

B	E	P(A=T)	P(A=F)
T	T	0.95	0.05
T	F	0.94	0.06
F	T	0.29	0.71
F	F	0.001	0.999

Conditional probability ~~of~~ table of Alarm A

A	$P(P_1=T)$	$P(P_1=F)$
T	0.90	0.10
F	0.05	0.95

C.P. ~~of~~ table of P_1 Calls

C.P. table of P_2 Calls.

A_1	$P(P_2=T)$	$P(P_2=F)$
T	0.70	0.30
F	0.01	0.99

* $P(P_1, P_2, A, \neg B, \neg E)$

$= P(P_1/A) \cdot P(P_2/A) \cdot P(A/\neg B, \neg E) \cdot P(\neg B) \cdot P(\neg E)$

$= 0.90 \times 0.70 \times 0.001 \times 0.999 \times 0.998$

$= 0.00062$

* The semantics of Bayesian N/w:-

① It is helpful to understand how to construct the N/w.

20 SUNDAY

② It is helpful in designing inference procedure.

* Hidden Markov Model :- HMM

- It is a statistical model which generates a model.
- Model generation based on some set of input sequence. $(S_1, S_2, \ldots S_n)$

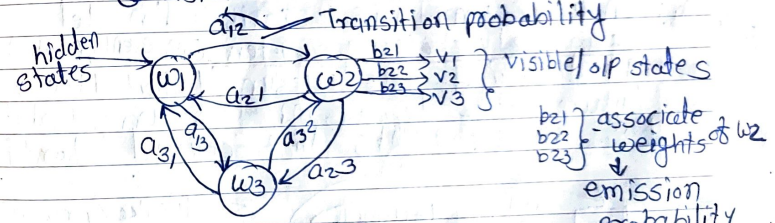
* Markov Property Satisfaction -

- It is a memory less.
  bcz, it is considering only 1 state at a time.
- & whenever u want to make current state this state depends on previous state.
- Sole part includes → future prediction → present state (past or future)

* HMM: It is same as like Finite State m/c's.

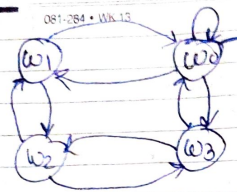
① Hidden state (ω)

② Visible state/output state (V)



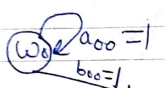
$\sum_j a_{ij} = 1$ $\quad$ $\sum_k b_{jk} = 1$

bcz, probability = is in betn 0 to 1 range



*final state

e.g.



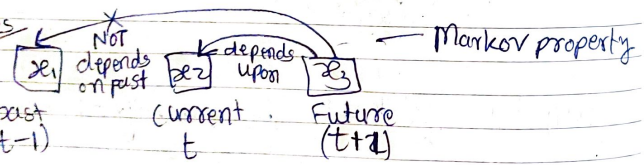
$a_{00} = 1$
$b_{00} = 1$
$\boxed{V_0}$

* Rinad Suman Academy | YouTube

* Markov Process :- OR Markov Model :-

wikipedi: A Markov model is a statistical model randomely changing system. (OR events, states) It is assumed that the future states depends on the current state, NOT on the events that occured before it.

states



— Markov property

* (t+1) event depends upon on t event not on t-1

$S = \{x_1, x_2, x_3\}$

- state space or set of states
- N distinct state $N=3$

e.g. ① Weather — Sunny - S, Rainy - R, Cloudy - C *

* Today's is Rainy, tommorrow it could be cloudy or it could be the sunny.

— So these s/m's are randomly changing s/m.

— If u want to predict tommorrow it will be cloudy or NOT, it could be depends upon today's wheather i.e. Rainy.

② Customer product — Nokia, Samsung, Iphone

③ Infant Baby — playing, eating, sleeping, crying

④ Tourist City visit — A city, B city, C city

⑤ Game Result — Loose, Draw, win.

* Markov property

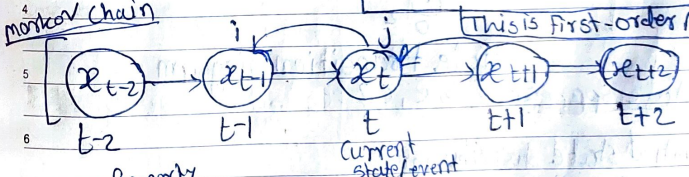
x_{t+1} depends on x_t & x_t depends on x_{t-1}

* if let $\{x_0, x_1, x_2, \ldots\}$ be a sequence of discrete random variable, then $\{x_0, x_1, x_2 \ldots\}$ is a Markov Chain, if it satisfies the Markov property.

$P(X_{t+1} = s | X_t = s_t, X_0 = s_0)$
$= P(X_{t+1} = s | X_t = s_t)$

Markov chain

This is first-order Markov Model.



Markov Property

$P(x_t | x_{t-1})$ & $P(x_{t+1} | x_t)$ => Conditional probability

$\therefore P_{ij} = P(X_t = j | X_{t-1} = i)$

* Any chain that follows the markov property that chain is called as Markov Chain.

Transition Matrix of Markov Chain:

Future state →

3×3 matrix

$$
\begin{array}{c|ccc}
 & S & R & C \\
\hline
S & 0.8 & 0.15 & 0.05 \\
R & 0.38 & 0.6 & 0.02 \\
C & 0.75 & 0.05 & 0.2
\end{array}
\quad
\begin{array}{l}
= 1 \text{ Row} \\
= 1 \\
= 1
\end{array}
$$

↓ current

Transition Probability $= P(X_{t+1} = j \mid X_t = i)$

e.g. Q① Given that today the weather is sunny (S), what is the probability that tommorrow is sunny (S) & day after is Raining (R).

Soln: $t_1 = S$, $t_2 = S$, $t_3 = R$

$P(t_3 = R, t_2 = S \mid t_1 = S) =$

$= P(t_3 = R \mid t_2 = S) \times P(t_2 = S \mid t_1 = S)$

$= 0.15 \times 0.8$

$= 0.120$

$= 12\%$

e.g. ② Given that today the weather is cloudy (C) & yesterday was Rainy (R) what is the probability that tommorrow would be sunny (S)?

⇒ $t_1 = R$ $t_2 = C$ $t_3 = S$

$P(t_3 = S \mid t_2 = C, t_1 = R) = P(t_3 = S \mid t_2 = C) \times P(t_2 = C \mid t_1 = R)$

$= 0.75 \times 0.02$

$= 0.0150$

$= 1.5\%$

eg Suppose u r sitting in room & u observe ur kids mood depending on observable states.

Hidden States [SU → RA → CL →]

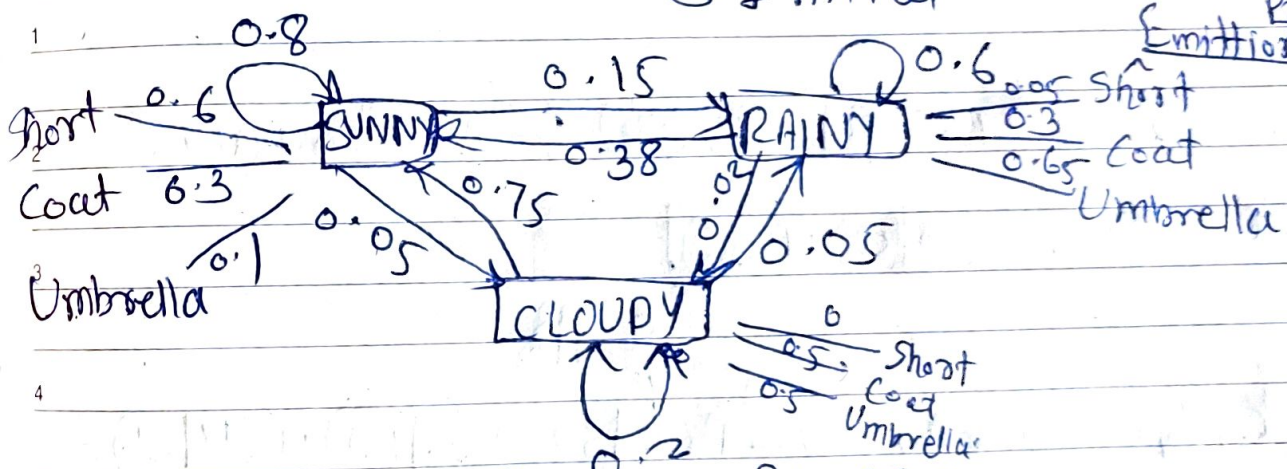
Observable States [playing, crying, quiet]

Dataset
- Training
- Testing data.

In HMM
1. π (Emmition prob.)
2. A (Transit^n prob.)
3. B (Emition prob.)

observable model → hidden model

HMM
- States
  1. Hidden
  2. Observable
- Probability
  1. Transition
  2. Emission
  3. Initial



Transit^n State Diagram

π [0.75, 0.2, 0.05]
(S, R)

* Transit^n matrix :-

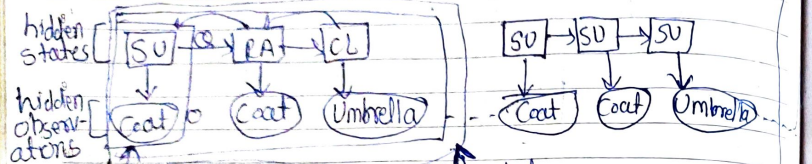
A	SU	RA	CL
SU	0.8	0.15	0.05
RA	0.38	0.6	0.02
CL	0.75	0.05	0.2

* Emittion matrix

B	Short	Coat	Umbrella
SU	0.6	0.3	0.1
RA	0.05	0.30	0.65
CL	0	0.5	0.5

joint prob. $= P(A|B)\,P(B)$

* HMM: Hidden Markov Model :-



So, ~~No~~ N = No. of diff. hidden states $= 3$

T = No. of observations $= 3$

$M = N^T = 3^3 = 27$ — possibilies of states

∴ 27 combinatns of hidden states.

$P(Coat, SU) = P(Coat|SU)\cdot P(SU)$ ← IMP formula.

$$\boxed{P(O,Q) = P(O|Q)\cdot P(Q)}$$

O Q

as HS

* $P(C, C, U, SU, RA, CL) = P(C|SU)\cdot P(C|RA)\cdot P(U|CL) \times$

Transition matrix $\boxed{P(SU)\cdot P(RA|SU)\cdot P(CL|RA)}$

Acc. Markov Prob. first weather

Emmission matrix $= 0.3 \times 0.30 \times 0.5 \times 0.75 \times 0.15 \times 0.02$

~~$0.75 \times 0.15 \times 0$~~ $= P_1$

$$\boxed{P(O,Q) = \therefore \prod P(O_i|q_i) \times \prod P(q_i|q_{i-1})}$$

$\therefore P_1 =$

$P_2 =$

$P_3 =$

⋮

$P_{27} =$

HMM $=$ $\max(P_1, P_2, P_3, \ldots P_{27})$

* Utility theory :-

* It is depending on 2 terms i.e.

1. Utility function & Rational Agent

- Rational agent means — e.g. Ideale Student. or Ideale person

- So, the utility theory means every agent ~~by its~~ become a rational agent by using utility theory.

- The agent should act like rational agent in which the rational agent would acted.

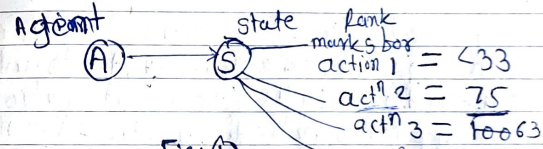


Fig: ①

* Agent It always try to act like rational agent. Agent on state always try to make rank 100%. gain & become rational agent.

* Maximum Expected Utility :- [MEU]

From fig ① when agent goes on state & sees the maximum utility of the state i.e. 75% i.e. the maximum expected utility.

Pankaj Porwal:

31 THURSDAY MARCH

* Model Based Reinforcement Learning Algorithm:

* Value Iteration:-

There are 2 common approaches to find optimal policy using recursive Bellman Equation in Model based algorithms.

1] Value Iteration:-

- In this method, the optimal policy (i.e. optimal action for a given state) is obtained by choosing the action that maximizes optimal state value funct^n for the given state.
- The optimal state function is obtained using an iterative algorithm, thats why this method is called value iteration.
- In this approach policy function is not used explicity during iteration.

2] Policy Iteration – In this method, optimal policy is obtained by finding better estimates of optimal policy function iteratively.

- Thats why this method is called policy iteration.
- There are 2 steps in this method:

i) Policy evaluation: In this step we evaluate the state value funct^n for current policy.

$\pi_0 \xrightarrow{E} V_0(s) \xrightarrow{I} \pi_1 \xrightarrow{E} V_1(s) \xrightarrow{I} \pi_2 \xrightarrow{E} V_2(s) \cdots \xrightarrow{I} \pi^* \xrightarrow{E} V^*(s)$

current policy — state value funct^n

FRIDAY APRIL 01

ii) Policy improvement: In this step we evaluate the state value funct^n for current policy function is improved by selecting the action that maximizes the state value function for each state.

* Algo. for Value Iteration:-

1] Set $V_0(s)$ for all state s.
optimal state value funct^{n}s

2] Iterate following until $V_i(s)$ converge to $V^*(s)$ for all states s:

$$V_{i+1}(s) = \max_{a} \sum_{s' \in S} T(s,a,s')\{r(s,a,s') + \gamma V_i(s')\}$$

Transition probability funct^n — Reward funct^n — gamma is numbers.

* Optimal Policy:-

$$a^* = \pi^*(s) = \arg\max_{a} \sum_{s' \in S} T(s,a,s')\{r(s,a,s') + \gamma V^*(s')\}$$

* Policy Iteration –

1) Initialize the optimal policy π_0 randomly
2) Iterate foll. until π_i converge to π^*

① Policy Evaluation:

$$V(s) = \sum_{s' \in S} T(s, \pi(s), s')[r(s, \pi(s), s') + \gamma V(s')] \quad -①$$

for all states s

transition probability function

reward function

(ii) Policy Improvement:-

$$\pi_{i+1}(s) = \underset{a}{\operatorname{argmax}} \sum_{s' \in S} T(s,a,s')[r(s,a,s') + \gamma V^*(s')]$$

gama - is number

eqn ① can be solved in 2 ways (i) as a sim of linear eqns in V(s) for all states s
(ii) iteratively, iterative soln here will be much easier then in value iteratn as the policy π_i at step i is known.

* Angular JS:-

Angular JS is an open source JS framework by Google to build web applications.

- Prerequisite: before learning Angular JS, you must hv the basic knowledge of JS.

- It is an excellent framework for building single phase appln & line of business appln s.

* Advantages of Angular JS:-
① Dependancy Injection:-
② Two way data binding -
③ Testing
④ Model View Controller.

③ Testing - Angular JS is designed in a way that we can test right from the start. so it is very easy to test any of its components through unit testing & end-to-end testing.

④ Model View Controller (MVC):-
In Angular JS, it is very easy to develop application in a clean MVC way.
You just hv to split your appln code into MVC components i.e. Model, View & the controller.