Unit-5

Knowledge
Inference

Knowledge Inference

- The knowledge base stored facts about the world. The inference engine applied logical rules to the knowledge base and deduced new knowledge. This process would iterate as each new fact in the knowledge base could trigger additional rules in the inference engine. Inference engines work primarily in one of two modes either special rule or facts:

- Forward chaining : Forward chaining starts with the known facts and asserts new facts.
- Backward chaining : Backward chaining starts with goals, and works backward to determine what facts must be asserted so that the goals can be achieved.

Horn Clause and Definite clause: This clause are the forms of sentences, which enables knowledge base to use a more restricted and efficient inference algorithm.

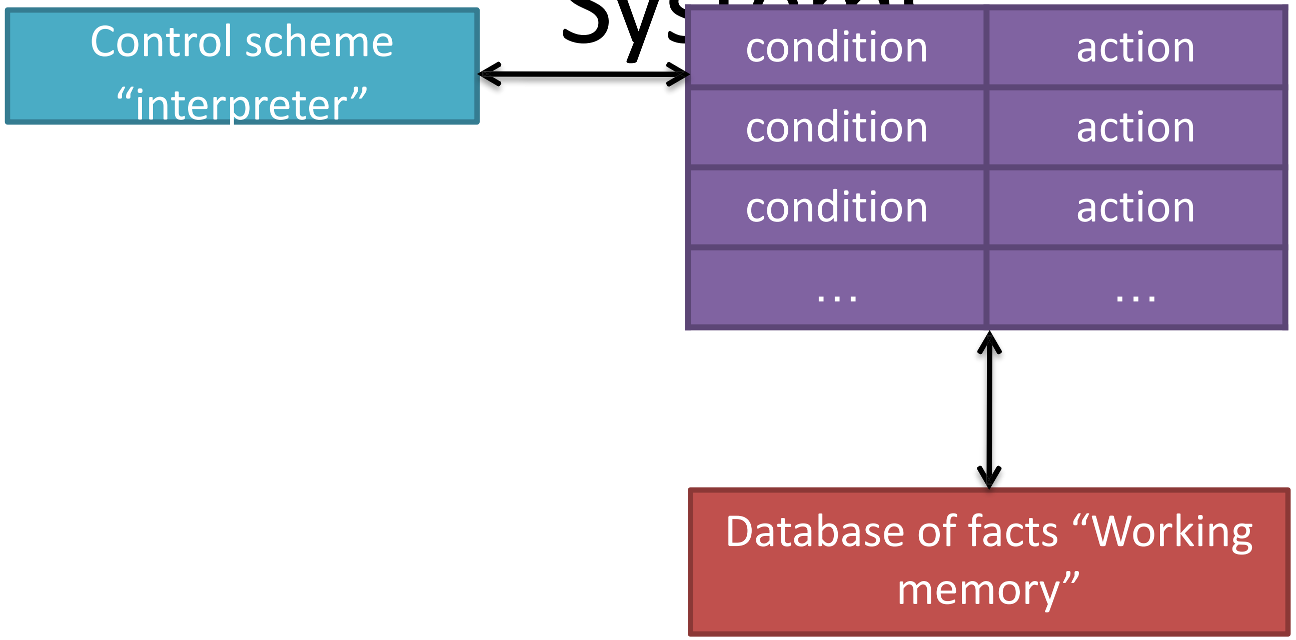
- **Definite clause:** A clause which is a disjunction of literals with **exactly one positive literal** is known as a definite clause or **strict horn clause**.
- **Horn clause:** A clause which is a disjunction of literals with **at most one positive literal** is known as horn clause. **Hence all the definite clauses are horn clauses**.

  - **Example: (¬ p V ¬ q V k)**. It has only one positive literal k. It is equivalent to p Λ q → k.

Conti.

- In propositional logic, resolution by forward/backward chaining
  - Forward: Start from knowledge to reach query
  - Backward: Start from query and go back
- In FOL, substitute variables to get propositions
  - Use lifting and unification to resolve variables
- Logic programming: Prolog, LISP, Haskell
- **Prolog**: Most widely used logic language.
  - Rules are written in backwards. Variables are uppercase and constants lowercase. Because of complexity, often compiled into other languages like: Warren Abstract Machine, LISP or C. Language makes it easy to contruct lists, like LISP.
- **LISP**: LISt Processing language: primary data structure is lists.
  - Lisp is used for AI because it can work with symbols. E.g: computer algebra, theorem proving, planning systems, diagnosis, rewrite systems, knowledge representation and reasoning, logic languages, machine translation, expert systems.
  - It is a functional programming language, as opposed to a procedural or imperative language

Rule-Based Systems



Rule n:
IF condition1
condition2
…
$\}$ antecedent

THEN
action1
action2
…
$\}$ consequent

- When one part of the IF portion matches a fact in working memory, the antecedent is SATISFIED. When all antecedents are satisfied, the rule is TRIGGERED. When the consequent of a rule is performed, the rule is FIRED.

Three Phases

1. Match phase
   - Match left side of rules (antecedents) with facts in working memory
   - Unification
2. Conflict resolution
   - Of all the rules that are triggered, decide which rule to fire.
   - Some strategies include:
     - Do not duplicate rule with same instantiated arguments twice
     - Prefer rules that refer to recently created WM elements
     - Prefer rules that are more specific (antecedents are more constraining)
     - Prefer Mammal(x) & Human(x) -> add Legs(x,2) over Mammal(x) -> add Legs(x,4)
   - If rules are ranked, fire according to ranking
3. Act phase
   - Add or delete facts to WM as specified by rule consequent

Forward Chaining

- When a new fact is added to the KB, for each rule such that ``matches" (unifies with) the premise If the other premises are known then add the conclusion to the KB and continue chaining
- Forward chaining is data-driven. It is also known as a **forward deduction or forward reasoning method** when using an inference engine.
- Forward chaining is a form of reasoning which start with atomic sentences in the knowledge base and applies inference rules (Modus Ponens) in the forward direction to extract more data until a goal is reached.
- The Forward-chaining algorithm starts from known facts, triggers all rules whose premises are satisfied, and add their conclusion to the known facts. This process repeats until the problem is solved.
- **E.g. A, A->B, B. A is the starting point. A->B represents a fact. This fact is used to achieve a decision B.**
- **E.g. Tom is running (A), If a person is running, he will sweat (A->B), Therefore, Tom is sweating. (B)**

Forward Chaining

Properties of Forward-Chaining:

- It is a process of making a conclusion based on known facts or data, by starting from the initial state and reaches the goal state.
- Forward chaining approach is also called as data-driven as we reach to the goal using available data.
- Forward chaining approach is commonly used in the expert system, such as CLIPS, business, planning, monitoring, control and interpretation application, automated inference engines, theorem proofs and production rule systems.

Forward Chaining

"As per the law, it is a crime for an American to sell weapons to hostile nations. Country A, an enemy of America, has some missiles, and all the missiles were sold to it by Robert, who is an American citizen."

Prove that **"Robert is criminal."**

First convert all the above facts into first-order definite clauses,

Facts Conversion into FOL:

- It is a crime for an American to sell weapons to hostile nations. (Let's say p, q, and r are variables)
  American (p) Λ weapon(q) Λ sells (p, q, r) Λ hostile(r) → Criminal(p) .....(1)
- Country A has some missiles. **there exist p Owns(A, p) Λ Missile(p)**. It can be written in two definite clauses by using Existential Instantiation, introducing new Constant T1.

  Owns(A, T1) …..(2)

  Missile(T1) …..(3)
- All of the missiles were sold to country A by Robert.
  forall p Missiles(p) Λ Owns (A, p) → Sells (Robert, p, A) .....(4)
- Missiles are weapons.
  Missile(p) → Weapons (p) ....(5)

Forward Chaining

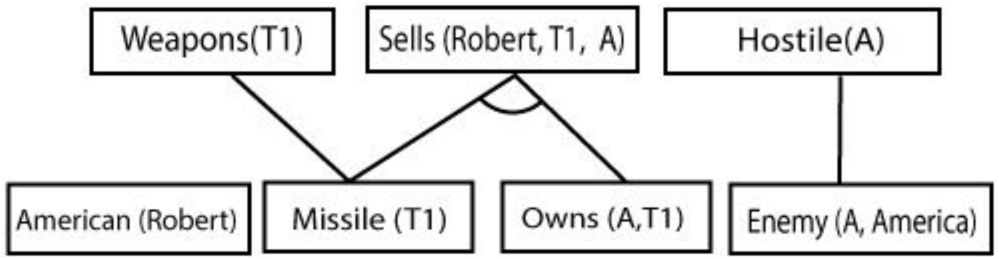
- Enemy of America is known as hostile.
  Enemy(p, America) →Hostile(p)   ....(6)
- Country A is an enemy of America.

  Enemy (A, America) ……..(7)
- Robert is American

  American(Robert). ……..(8)

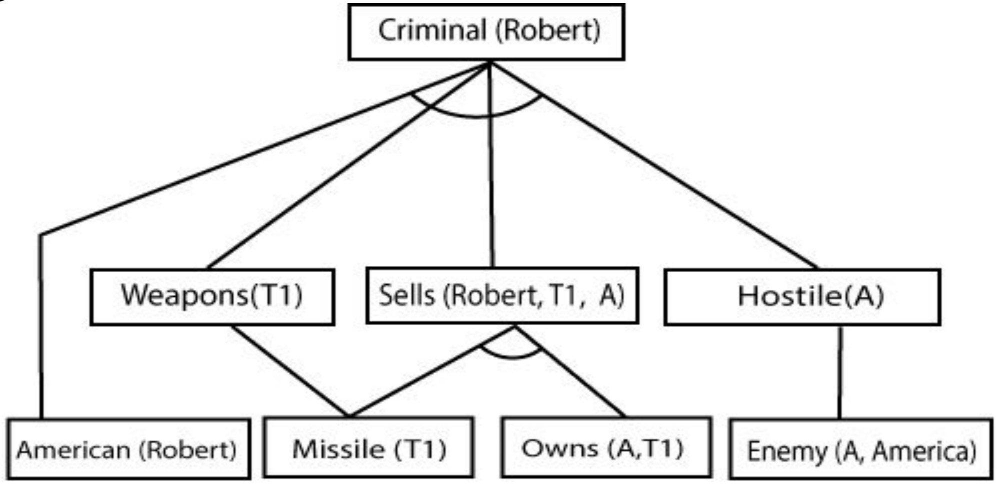
Forward Chaining

- **Step-1:**
- In the first step we will start with the known facts and will choose the sentences which do not have implications, such as: **American(Robert), Enemy(A, America), Owns(A, T1), and Missile(T1)**.
- **Step-2:**
- At the second step, we will see those facts which infer from available facts and with satisfied premises.
- Rule-(1) does not satisfy premises, so it will not be added in the first iteration.
- Rule-(2) and (3) are already added.
- Rule-(4) satisfy with the substitution {p/T1}, **so Sells (Robert, T1, A)** is added, which infers from the conjunction of Rule (2) and (3).
- Rule-(6) is satisfied with the substitution(p/A), so Hostile(A) is added and which infers from Rule-(7).



Forward Chaining

Step-3: At Step-3, as we can check Rule-(1) is satisfied with the substitution **{p/Robert, q/T1, r/A}, so we can add Criminal(Robert)** which infers all the available facts. And hence we reached our goal statement.



Hence it is proved that Robert is Criminal using forward chaining approach.

Forward Chaining

Advantages

- It can be used to draw multiple conclusions.
- It provides a good basis for arriving at conclusions.
- It’s more flexible than backward chaining because it does not have a limitation on the data derived from it.

Disadvantages

- The process of forward chaining may be time-consuming. It may take a lot of time to eliminate and synchronize available data.
- Unlike backward chaining, the explanation of facts or observations for this type of chaining is not very clear. The former uses a goal-driven method that arrives at conclusions efficiently.

Forward Chaining

There are two main types of control schemes that are applied to rule-based systems.

Z1 If ?x has hair
Then ?x is a mammal
Z2 If ?x gives milk
Then ?x is a mammal
Z3 If ?x has feathers
Then ?x is a bird
Z6 If ?x is a mammal
?x has pointed teeth
?x has claws
?x has forward-pointing eyes
Then ?x is a carnivore
Z8 If ?x is a mammal
?x chews cud
Then ?x is an ungulate
Z11 If ?x is an ungulate
?x has long legs
?x has long neck
?x has tawny color
?x has dark spots
Then ?x is a giraffe

Database

F1) Stretch has hair
F2) Stretch chews cud
F3) Stretch has long legs
F4) Stretch has a long neck
F5) Stretch has tawny color
F6) Stretch has dark spots

Forward Chaining

- Reason FORWARD from facts/rules to (hopefully) a needed goal
- Use modus ponens to generate new facts
- Rule antecedents are compared with facts from database
- If match, add consequents to database
- Repeat as long as needed
- Forward chaining is “data driven”

Backward Chaining

- Backward-chaining is also known as a **backward deduction or backward reasoning method** when using an inference engine.
- A backward chaining algorithm is a form of reasoning, which starts with the goal and works backward, chaining through rules to find known facts that support the goal.
- Backward chaining can be used in debugging, diagnostics, prescription applications, game theory, automated theorem proving tools, inference engines, proof assistants, and various AI applications.
- When a query q is asked
  - if a matching fact q' is known, return the unifier
  - for each rule whose consequent q' matches q
  - attempt to prove each premise of the rule by backward chaining
- Two versions: find **any** solution, find **all** solutions
- Backward chaining is the basis for **logic programming**, e.g., Prolog

Backward Chaining

- **Properties of backward chaining:**
- In backward chaining, the goal is **broken into sub-goal or sub-goals to prove the facts true**. It is called a **goal-driven approach**, as a list of goals decides which rules are selected and used.
- The **modus ponens inference rule** is used as the basis for the backward chaining process. This rule states that if both the conditional statement (p->q) and the antecedent (p) are true, then we can infer the subsequent (q).
- Use modus ponens
  - Start at goals
  - Match goals to consequents or facts
  - If match consequents, antecedents become new sub goals
- Repeat until
  - All sub goals are proven or
  - At least one sub goal cannot be proven

Backward Chaining

- Backward chaining can be explained in the following sequence.
- B, A->B, A.
- B is the goal or endpoint, that is used as the starting point for backward tracking. A is the initial state. A->B is a fact that must be asserted to arrive at the endpoint B.
- E.g. Tom is sweating (B), If a person is running, he will sweat (A->B), Tom is running (A).

Advantages

- The result is already known, which makes it easy to deduce inferences.
- It’s a quicker method of reasoning than forward chaining because the endpoint is available.
- In this type of chaining, correct solutions can be derived effectively if pre-determined rules are met by the inference engine.

Disadvantages

- The process of reasoning can only start if the endpoint is known.
- It doesn’t deduce multiple solutions or answers.
- It only derives data that is needed, which makes it less flexible than forward chaining.

Backward Chaining

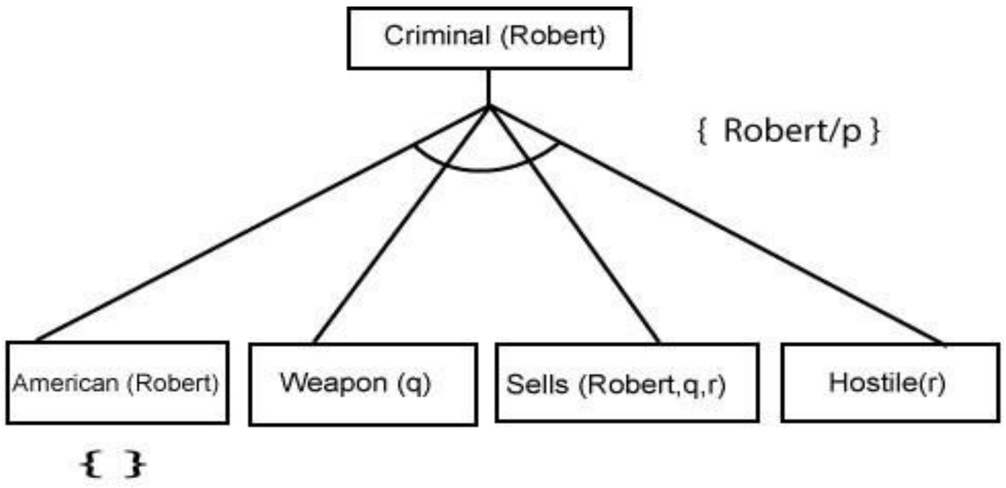
- The backward-chaining method mostly used a **depth-first search** strategy for proof.
- **Example**:
- In backward-chaining, we will use the same above example, and will rewrite all the rules.
- **American (p) A weapon(q) A sells (p, q, r) A hostile(r) → Criminal(p) ...(1)**
  Owns(A, T1) ....(2)
- **Missile(T1)**

......(4)

- **Forall p Missiles(p) A Owns (A, p) → Sells (Robert, p, A)**
- **Missile(p) → Weapons (p) ....(5)**
- **Enemy(p, America) →Hostile(p) ....(6)**
- **Enemy (A, America) ....(7)**
- **American(Robert). ....(8)**

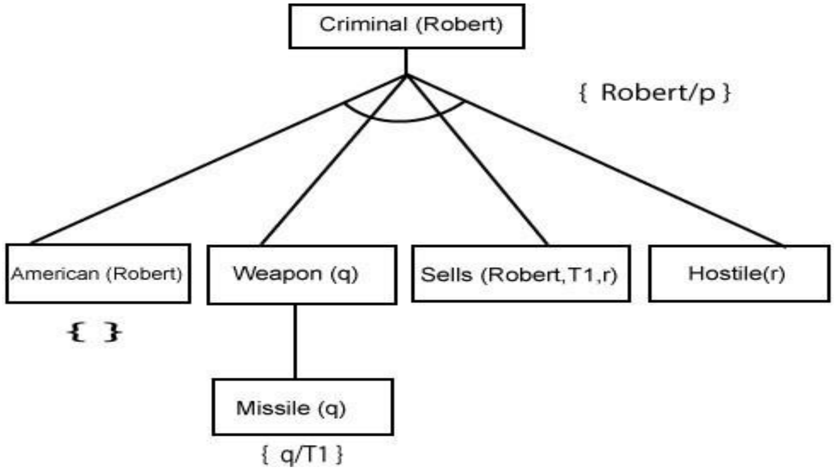
Backward Chaining

- Backward-Chaining proof:- In Backward chaining, we will start with our goal predicate, which is **Criminal(Robert)**, and then infer further rules.
- **Step-1:** At the first step, we will take the goal fact. And from the goal fact, we will infer other facts, and at last, we will prove those facts true. So our goal fact is "Robert is Criminal," so following is the predicate of it.
- **Step-2:** At the second step, we will infer other facts from goal fact which satisfies the rules. So as we can see in Rule-1, the goal predicate Criminal(Robert) is present with substitution {Robert/P}. So we will add all the conjunctive facts below the first level and will replace p with Robert.

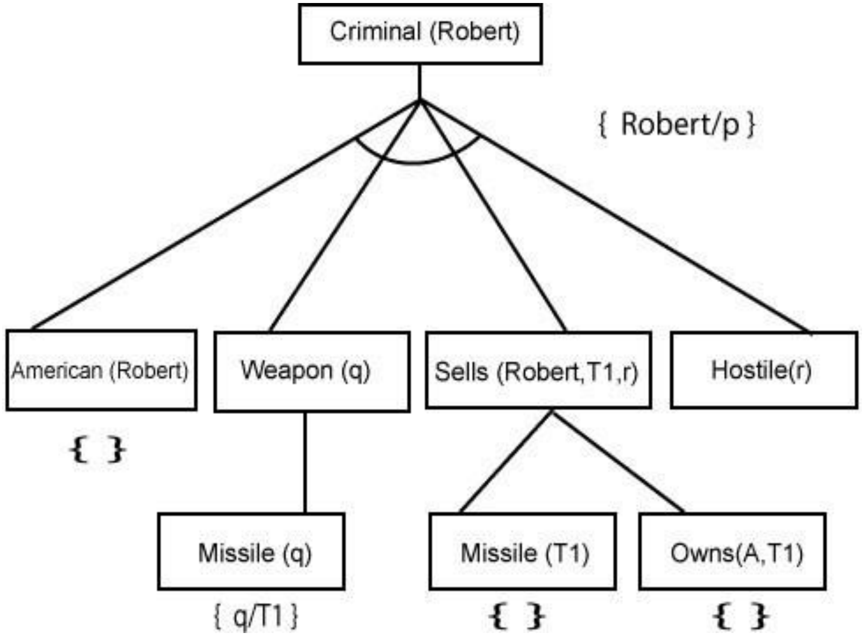


Backward Chaining

- **Step-3:** At step-3, we will extract further fact Missile(q) which infer from Weapon(q), as it satisfies Rule-(5). Weapon (q) is also true with the substitution of a constant T1 at q.

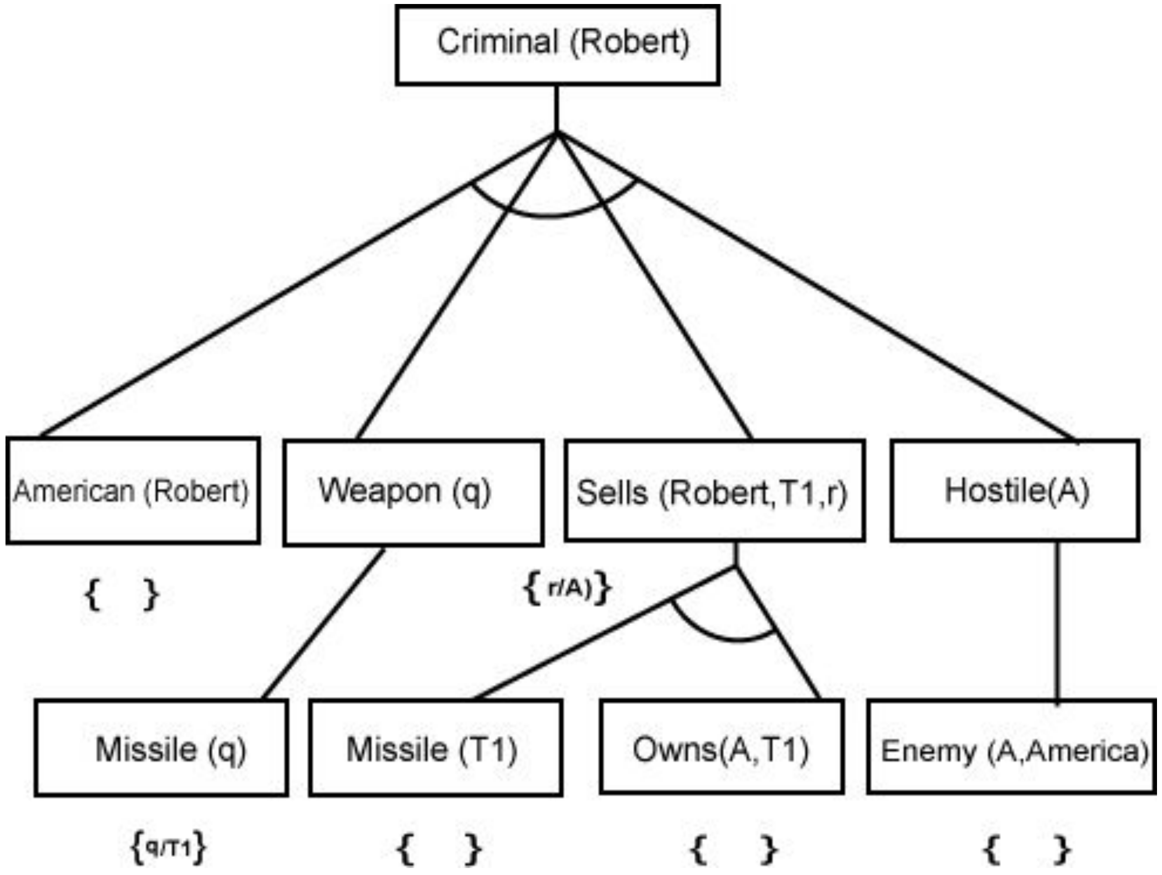


- **Step-4:** At step-4, we can infer facts Missile(T1) and Owns(A, T1) from Sells(Robert, T1, r) which satisfies the **Rule- 4**, with the substitution of A in place of r. So these two statements are proved here.



Backward Chaining

- **Step-5:** At step-5, we can infer the fact **Enemy(A, America)** from **Hostile(A)** which satisfies Rule-6. And hence all the statements are proved true using backward chaining.



Forward Chaining vs. Backward Chaining

Forward Chaining	Backward Chaining
It starts from known facts and applies inference rule to extract more data unit it reaches to the goal.	It starts from the goal and works backward through inference rules to find the required facts that support the goal.
It is a top down approach.	It is a bottom-up approach.
It is known as data-driven inference technique as we reach to the goal using the available data.	It is known as goal-driven technique as we start from the goal and divide into sub-goal to extract the facts.
It applies a breadth-first search strategy.	It applies a depth-first search strategy
It tests for all the available rules	It tests only for few required rules.
It is suitable for the planning, monitoring, control, and interpretation application.	It is suitable for diagnostic, prescription, and debugging application.
It can generate an infinite number of possible conclusions.	It generates a finite number of possible conclusions.

Reasoning System

- The reasoning is the mental process of deriving logical conclusion and making predictions from available knowledge, facts, and beliefs or we can say, "**Reasoning is a way to infer facts from existing data**." It is a general process of thinking rationally, to find valid conclusions.
- **Human reasoning** capability are divided into three areas:
  - **Mathematical reasoning**-axioms, definitions, theorems, proofs
  - **Logical reasoning** – deductive, inductive, abductive
  - **Non-logical reasoning** – linguistic, language
- These 3 areas of reasoning are in every human being but the **ability level depends on education, environment and genetics.**
- **Approaches to reasoning- Symbolic, statistical and fuzzy logic reasoning**
- To help machines perform human-level functions, AI and its various subfields rely on reasoning and knowledge representation. **Reasoning in AI helps machines think rationally and perform functions like humans**. Helps machines in **problem-solving, deriving logical solutions, and making predictions based on the available information, knowledge, facts, and data**.
- Additionally, reasoning can be **performed** either in a **formal and informal** manner or **top-down** or **bottom-up** approaches, depending on the way machines handle uncertainties and partial truths. Like Probabilistic Reasoning in AI allows machines to deal with and represent uncertain knowledge and information.

Role of Reasoning in Knowledge-based Systems

- Among the various types of reasoning, **deductive reasoning and non-monotonic reasoning are used by knowledge-based systems to solve problems**. Moreover, it helps with the implementation of knowledge-based systems and AI in machines, **enabling them to perform tasks that require human level intelligence and mental processes.**

- In short, like humans, machines use reasoning, along with knowledge representation, logic, and learning for analysis, problem-solving, making conclusions, and more.

- **Properties of Information**:
  - **Complete**: **All the facts** that are necessary to solve a problem are **present** in the system. E.g., Question Answering or proof.
  - **Consistent**: There is **no contradiction** in the available information. E.g., we **should not** be able to **derive both P and ~P.**
  - **Monotonous / Monotonically Growing**: An **addition of information does not make existing information false, or inconsistent.**

- **When the information is not complete, or inconsistent, we need Non-Monotonic Reasoning Systems.**

Uncertainty

- The world is an uncertain place, often the **knowledge is imperfect which causes uncertainty**. Therefore reasoning must be able to operate under uncertainty.

- **Uncertain inputs**
  - Missing data
  - Noisy data

- **Uncertain knowledge**
  - Multiple causes lead to multiple effects
  - Incomplete enumeration of conditions or effects
  - Incomplete knowledge of causality in the domain
  - Probabilistic effects

- **Uncertain outputs**
  - Abduction and induction are inherently uncertain
  - Default reasoning, even in deductive fashion, is uncertain
  - Incomplete deductive inference may be uncertain

Reasoning under uncertainty

- The reasoning is the mental process of deriving logical conclusion and making predictions from available knowledge, facts, and beliefs. Or we can say, "**Reasoning is a way to infer facts from existing data**." It is a general process of thinking rationally, to find valid conclusions.
- **Types of Reasoning**
  - Monotonic Reasoning
  - Non Monotonic Reasoning
  - Common Sense Reasoning
  - Deductive reasoning
  - Inductive reasoning
  - Abductive reasoning

Monotonic Reasoning

- In monotonic reasoning, **once the conclusion is taken, then it will remain the same even if we add some other information to existing information in our knowledge base**. In monotonic reasoning, adding knowledge does not decrease the set of prepositions that can be derived.
- To solve monotonic problems, **we can derive the valid conclusion from the available facts only, and it will not be affected by new facts**.
- Monotonic reasoning **is not useful for the real-time systems**, as in real time, facts get changed, so we cannot use monotonic reasoning.
- Monotonic reasoning is used in conventional reasoning systems, and a logic-based system is monotonic.
- **Any theorem proving is an example of monotonic reasoning**.
- **Example:**
  - **Earth revolves around the Sun.**
- It is a true fact, and it cannot be changed even if we add another sentence in knowledge base like, "The moon revolves around the earth" Or "Earth is not round," etc.

Monotonous Information

- Conventional reasoning system works with information that is:
  - Complete
  - Consistent
  - Monotonous.
- When do you say that the information is monotonous?
  - If a new fact gets added to the already existing information, and still, all the information remains the same, it does not change, consistency still remains across all the facts, and no fact has to be retracted, then, this information is said to be monotonous.

Monotonic Reasoning

- Standard type of logic
  - If proven true, will be true forever
  - Facts provided can't be modified
- Doesn't always fit in real life.
- Sidra is in Doha and Doha is in Qatar, so Sidra is in Qatar.
- Sidra can always take a trip to United States

Non Monotonic Reasoning

- In Non-monotonic reasoning, **some conclusions may be invalidated if we add some more information to our knowledge base.** Non-monotonic reasoning **deals with incomplete and uncertain models**. "**Human perceptions for various things in daily life**, "is a general example of non-monotonic reasoning.

- **Example:** Let suppose the knowledge base contains the following knowledge:
  - **Birds can fly**
  - **Penguins cannot fly**
  - **Pitty is a bird**

- So from the above sentences, we can conclude that **Pitty can fly**. However, if we add one another sentence into knowledge base "**Pitty is a penguin**", which concludes "**Pitty cannot fly**", so it invalidates the above conclusion.

Non Monotonic Reasoning

- A logic is **non-monotonic** if some conclusions can be invalidated by adding more knowledge. The logic of definite clauses with negation as failure is non-monotonic. Non-monotonic reasoning is useful for representing defaults. A default is a rule that can be used unless it overridden by an exception.
- New facts can be added
- Current facts can be modified
- Conclusion can change
  - If A $\rightarrow$ B before new fact
  - Conclusion might change after new fact
- **Example**
  - All balls bounce
  - Football is a ball
  - Does football bounce (Of course?)

- What about a football with no air filled in? (Conclusions change with new facts)

Non Monotonic Reasoning

Real life usage : Used in artificial intelligent systems

- For its adaptability
- Adding, removing and modifying facts
- To reach appropriate conclusions for appropriate scenarios

Application : Consider an example that can't be handled by monotonic logic

- Birds can fly (Seems logical, right?)

Non-monotonic Reasoning

- Non-monotonic reasoning can be describe using default reasoning.
- Default reasoning – The conclusions are drawn based on what is most likely to be true.
- There are **2 approaches**:
  - **Non monotonic logic**
  - **Default logic**
- **Non monotonic logic**: The **truth of a proposition may change when new information are added and a logic may be build to allows the statement to be retraced.**
- Non monotonic logic is predicate logic with one extension called **model operator(M)** which means "**consistent with everything we know**".
- The purpose of M is to allow consistency.

Default logic

- Logic base system is not able to handle non monotonous information

- Standard Logic → true or false
- Consider a rule "Birds can fly."
- Default Logic can formalise rules without explicitly mentioning all their exceptions.
- The rule can be formalized by the following 'default':

Precondition **Justification**

$$D = \left\{ \frac{Bird\ (X): Flies\ (X)}{Flies\ (X)} \right\}$$

Conclusion

This rule says that, "If X is a bird, and it can be assumed that it flies, then we can conclude that it flies."

- Consider the following is the knowledge that we have:

$$W = \{Bird(Pigeon), Flies(Pigeon), Bird(Penguin), \neg Flies(Penguine), Flies(Bee)\}$$

- Consider X as pigeon. We can conclude that a pigeon flies.
- Now, consider X is a penguin. Flies(penguin) is not consistent with what is known. That is why, you cannot conclude that penguin can fly.
- Now consider that X is a Bee.
- Default rule allows a delivery from the precondition to the justification and not from the justification to the precondition. Since for a bee the precondition is not satisfied, you cannot make a conclusion that the bee flies.

Default logic

- A default theory is a pair <W, D>, where W is a set of logical formulas called the background theory, that formalizes the facts that are known for sure and D is a set of default rules, also called 'Defaults' each one being of the form:

$$\frac{Prerequisite : Justification_1, \ldots, Justification_n}{Conclusion}$$

- It says that, if we believe that prerequisite is true, and each of the justifications is consistent with our current beliefs, we are led to believe that conclusion is true.

- For the ABC murder story, we can have a default like the following:

$$D = \left\{\frac{Accused\ (X): Innocent\ (X)}{Innocent\ (X)}\right\}$$

- Which says that if X is the accused and it is consistent to assume that he is innocent, then we can conclude that he is innocent.

Non Monotonic Inference

- **Non-monotonic reasoning**:
  - An approach to reason with incomplete information.
  - Add axioms/ rules to form an extended knowledge base.
  - Extended knowledge base is obtained using the existing knowledge and our new information obtained using non-monotonic reasoning.
  - An inference based on lack of knowledge is called Non-Monotonic Inference.
- Consider the following Knowledge Base:
  - Pooja likes ice cream
  - Pallavi likes ice cream.
- Does Ravina like ice cream?
  - The database does not have that information
  - The answer could be, false, Ravina does not like ice cream. → Non-Monotonic Inference

Closed World Assumption

- **Closed World Assumption** is an approach for dealing with incompleteness by assuming that, anything that is not contained within the knowledge base is false.
- Let us consider another example where the knowledge base consists of 5 cities, A, B, C, D and E, and the following information is available:
  - City A is connected to city B
  - City B is connected to city C
  - City C is connected to city D
  - City D is connected to city E
- Then the extended knowledge base will consists of statements that mention which is not connected to which city.
- Is city A connected to city E? → No!
- Our extended database says that, city A is not connected to city E.
- This would be the Non-Monotonic Inference.

Limitations of Closed World Assumption

1. Knowledge is not realistically "closable".
   - For example in the ABC Murder story W
     - A or B or C committed the crime
     - A did not commit the crime
     - B did not commit the crime
   - This was our closed world. C's proof was not present in the knowledge base.
2. Syntax problem: Conjunction can be handled by CWA. But not Disjunction.

Knowledge Base: single(Jim) ∨ single(Pam)	
Is Jim single? →The answer cannot be yes	Is Pam single? →The answer cannot be yes
¬single(Jim)	¬single(Pam)

   - So the extended database would look like this
     - Single(Jim) V Single(Pam)
     - ¬ single(Jim)
     - ¬ single(Pam)
   - This extended data base is inconsistent!
   - This example shows that **Disjunctions cannot be handled by the Closed World Assumption.**

Generalized Closed World Assumption

- The Generalised CWA allows addition of statements only if it does not make the existing knowledge base inconsistent.

Knowledge Base: single(Jim) v single(Pam)	
Is Jim single? →The answer cannot be yes	Is Pam single? →The answer cannot be yes
¬single(Jim)	¬single(Pam)

- So the extended Knowledge base using GCWA looks like this:
  - Single(Jim) V Single(Pam)
  - ¬single(Jim)
- This keeps the knowledge base consistent. But not necessarily complete.

Common sense Reasoning

- Common sense reasoning is an **informal form of reasoning**, which can be **gained through experiences.**
- Common Sense reasoning simulates the human ability to make presumptions about events which occurs on every day.
- It relies on **good judgment rather than exact logic** and operates on **heuristic knowledge** and **heuristic rules**.
- **Example:**
  - **One person can be at one place at a time.**
  - **If I put my hand in a fire, then it will burn.**
- The above two statements are the examples of common sense reasoning which a human mind can easily understand and assume.
- It is mainly used in computer vision and robotic manipulation.

Deductive Reasoning

- Deductive reasoning is **deducing new information from logically related known information**. It is the **form of valid reasoning**, which means the argument's conclusion must be true when the premises are true.
- Deductive reasoning is a **type of propositional logic** in AI, and it requires various rules and facts. It is sometimes referred to as **top-down reasoning**, and contradictory to inductive reasoning.
- In deductive reasoning, the truth of the premises guarantees the truth of the conclusion.
- Deductive reasoning mostly starts from the **general premises to the specific conclusion**, which can be explained as below example.
- **Example:**
  - **Premise-1: All the human eats veggies**
  - **Premise-2: Suresh is human.**
  - **Conclusion: Suresh eats veggies.**

Deductive Reasoning

- Deductive reasoning **uses formal logic** to produce logically certain results, using a set process, which involves the following steps.
  - Theory
  - Hypothesis
  - Patterns
  - Confirmation

Inductive Reasoning

- An opposite of deductive reasoning, inductive reasoning is a **bottom-up logic** that **uses specific observations to reach a conclusion**.
- Used in cases where there is a limited set of facts and data to arrive at a conclusion, Inductive reasoning **uses historical data to produce a generic rule** whose conclusion is supported by the premises.
- Also **known as cause-effect reasoning**, this second type of reasoning is exploratory in nature and allows for uncertain but likely results.
- Inductive reasoning follows a set of steps to perform reasoning, these are:
  - Observation
  - Pattern
  - Hypothesis
  - Theory

Inductive vs deductive reasoning

- Deductive and inductive reasoning are opposites — **deduction applies a top-to-bottom**(general to specific) approach to reasoning whereas **induction applies a bottom-to-top**(specific to general) approach.
- **Inductive Reasoning** – Specific to general
  - Logically true
  - May or may not be realistically true (But not definitely true)
  - **Inductive reasoning** frequently used in mathematics by observing patterns that exist in a particular case; we deduce the general conclusion from that outcome. It is looking for a pattern or looking for a trend and then generalizing.
  - The conclusion that we arrive based on inductive reasoning is called as **conjecture**. Conjecture is a hypothesis that has not be proven. Just because we observe a pattern in many cases does not mean it's true for all cases. Conjecture must be prove for that particular case. To prove such conjecture, principal of mathematical induction is used.
  - Inductive reasoning starts with a specific assumption, then it broadens in scope until it reaches a generalized conclusion. With inductive reasoning, the conclusion may be false even if the premises are true.

Inductive vs deductive reasoning

- **Deductive Reasoning** – General argument to specific conclusion
  - Logically true
  - Realistically true (Always true)

Inductive vs deductive reasoning

- **Inductive Reasoning**: The first lipstick I pulled from my bag is red. The second lipstick I pulled from my bag is red. Therefore, all the lipsticks in my bag are red.

  Deductive Reasoning: The first lipstick I pulled from my bag is red. All lipsticks in my bag are red. Therefore, the second lipstick I pull from my bag will be red, too.

- **Inductive Reasoning**: My mother is Irish. She has blond hair. Therefore, everyone from Ireland has blond hair.

  Deductive Reasoning: My mother is Irish. Everyone from Ireland has blond hair. Therefore, my mother has blond hair.

- **Inductive Reasoning**: Maximilian is a shelter dog. He is happy. All shelter dogs are happy.

  Deductive Reasoning: Maximillian is a shelter dog. All shelter dogs are happy. Therefore, he is happy.

Inductive vs. Deductive Reasoning

- Inductive and deductive reasoning are the two most important and commonly used reasoning techniques.

INDUCTIVE REASONING	DEDUCTIVE REASONING
This type of reasoning reaches the conclusion through the process of generalization using facts and data.	Type of valid reasoning, where new information or conclusion is deduced using known facts and information.
It follows a bottom-up approach and starts from the conclusion.	It follows a top-down approach and starts from the premises.
The conclusion is not guaranteed to be true if the premises are true.	Here the conclusion is true if the premises are true.
It is fast and easy, as it requires evidence instead of facts.	Compared to inductive reasoning, it is difficult to use as it requires true facts.

Inductive research approach

- When there is little to no existing literature on a topic, it is common to perform inductive research because there is no theory to test. The inductive approach consists of three stages:
- **Observation**
  - A low-cost airline flight is delayed
  - Dogs A and B have fleas
  - Elephants depend on water to exist
- **Observe a pattern**
  - Another 20 flights from low-cost airlines are delayed
  - All observed dogs have fleas
  - All observed animals depend on water to exist
- **Develop a theory**
  - Low cost airlines always have delays
  - All dogs have fleas
  - All biological life depends on water to exist
- **Limitations of an inductive approach**
  - A conclusion drawn on the basis of an inductive method can never be proven, but it can be invalidated.

Deductive research approach

- When conducting deductive research, you always start with a theory (the result of inductive research). Reasoning deductively means testing these theories. If there is no theory yet, you cannot conduct deductive research. The deductive research approach consists of four stages:

- **Start with an existing theory**
  - Low cost airlines always have delays
  - All dogs have fleas
  - All biological life depends on water to exist
- **Formulate a hypothesis based on existing theory**
  - If passengers fly with a low cost airline, then they will always experience delays
  - All pet dogs in my apartment building have fleas
  - All land mammals depend on water to exist

Deductive research approach

- **Collect data to test the hypothesis**
  - Collect flight data of low-cost airlines
  - Test all dogs in the building for fleas
  - Study all land mammal species to see if they depend on water
- **Analyse the results: does the data reject or support the hypothesis?**
  - 5 out of 100 flights of low-cost airlines are not delayed = reject hypothesis
  - 10 out of 20 dogs didn't have fleas = reject hypothesis
  - All land mammal species depend on water = support hypothesis

Deductive research approach

- **Limitations of a deductive approach**

  The conclusions of deductive reasoning can only be true if all the premises set in the inductive study are true and the terms are clear.

  - **Example**
    - All dogs have fleas (premise)
    - Benno is a dog (premise)
    - Benno has fleas (conclusion)
- Based on the premises we have, the conclusion must be true. However, if the first premise turns out to be false, the conclusion that Benno has fleas cannot be relied upon.

Abductive Reasoning

- An extension of deductive reasoning, abductive reasoning, is a **form of logical inference that seeks theories to define observations using the simplest and likely explanation**.
- Compared to inductive reasoning, abductive reasoning, which is also known as **abductive inference or retroduction**, is **less rigorous and allows for most accurate suggestions and guesses**.
- However, unlike deductive reasoning, it **does not rectify the plausible conclusion and is mostly used in data certainties**. It is associated with troubleshooting and decision making.
- Abductive reasoning is a form of logical reasoning which starts with single or multiple observations then seeks to find the most likely explanation or conclusion for the observation. Abductive reasoning is an extension of deductive reasoning, but in **abductive reasoning, the premises do not guarantee the conclusion.**
- Example:

  Incomplete observations -> Best Prediction (may be true)

  – Implication: Cricket ground is wet if it is raining

  – Axiom: Cricket ground is wet.

  – Conclusion It is raining.

Probabilistic Reasoning

- Probabilistic reasoning is a **way of knowledge representation** where we **apply the concept of probability to indicate the uncertainty in knowledge**. In probabilistic reasoning, we **combine probability theory with logic to handle the uncertainty.**
- We use probability in probabilistic reasoning because it provides a way to handle the uncertainty that is the result of someone's laziness and ignorance.
- In the real world, there are lots of scenarios, where the **certainty of something is not confirmed**, such as "It will rain today," "behaviour of someone for some situations," "A match between two teams or two players." These are probable sentences for which we can assume that it will happen but not sure about it, so here we use probabilistic reasoning.

Need of probabilistic reasoning in AI:

- When there are unpredictable outcomes.
- When specifications or possibilities of predicates becomes too large to handle.
- When an unknown error occurs during an experiment.
- In probabilistic reasoning, there are two ways to solve problems with uncertain knowledge:
  - **Bayes' rule**
  - **Bayesian Statistics**

Bayes' Theorem

- P(H | E)

Hypothesis Evidence

- Truthiness is depend on how many evidence supporting the hypothesis.
- P(Hi | E) – What is probability of hypothesis that evidence is available.
- P(E | Hi) – Probability of evidence being present for a particular hypothesis.
- P(Hi) – Probability for ith hypothesis to true (Whatever evidence available or not)
- Bayes' theorem:

$$P(H_i \mid E) = P(E \mid H_i) * P(H_i) / \sum P(E \mid H_n) * P(H_n) \text{ , where } n=1 \text{ to } k.$$

- Application of Baye's theorem in AI
  - Next step is calculated based on prior step
    - Robot
    - Automatic Machine
  - Forecasting
    - Weather

Bayes'

BAYE'S THEOREM: Describes the probability of an event, based on prior knowledge of conditions that might be related to the event.

↳ In Probability theory it relates the Conditional probability & marginal probabilities of two random events.

↳ Calculate $P(B|A)$ with Knowledge of $P(A|B)$.

↳ $P(H|E) = \dfrac{\text{no. of time H and E}}{\text{no. of times E}}$

$P(H|E) = \dfrac{P(H \cap E)}{P(E)}$ } Prob. of H when E is true.

$P(A \cap B) = P(A|B).P(B)$ — (i)
$P(A \cap B) = P(B|A).P(A)$ — (ii)
} From (i) and (ii) L.H.S are equal.

↳ $P(A|B).P(B) = P(B|A).P(A)$

So, $\boxed{P(B|A) = \dfrac{P(A|B).P(B)}{P(A)}}$ Baye's theorem formula.

Posterior (Prob. of A when B is true)

marginal Prob. (Prob. of evidence)

Likelihood (Prob. of evidence)

Prior Prob (Prob. of hypothesis)

Bayes'

Baye's theorem Example 1:

Ques 1:- what is the probability that Person has disease dengue with neck pain?

↳ Given:-

→ 80% of time dengue causes neck pain. $P(a|b) = 0.8$

↳ $P(\text{dengue}) = 1/30{,}000$ $(P(b) \Rightarrow 1/30{,}000$

↳ $P(\text{neckpain}) = .02$ $P(a) = .02$

a = Proposistion that Person has neck pain

b = Person has dengue.

$P(b|a) = ?$

$$P(b|a) = \frac{P(a|b) \cdot P(b)}{P(a)}$$

$$= \frac{0.8 \cdot 1/30000}{0.02} = \boxed{0.00133}$$

Probabilistic Reasoning

- As probabilistic reasoning uses probability and related terms, so before understanding probabilistic reasoning, let's understand some common terms:

  Probability: Probability can be defined as a chance that an uncertain event will occur. It is the numerical measure of the likelihood that an event will occur. The value of probability always remains between 0 and 1 that represent ideal uncertainties.

  - $0 \leq P(A) \leq 1$, where $P(A)$ is the probability of an event A.
  - $P(A) = 0$, indicates total uncertainty in an event A.
  - $P(A) = 1$, indicates total certainty in an event A.

Probabilistic Reasoning

- We can find the probability of an uncertain event by using the below formula.

$$\text{Probability of occurrence} = \frac{\text{Number of desired outcomes}}{\text{Total number of outcomes}}$$

- P(¬A) = probability of a not happening event.
- P(¬A) + P(A) = 1.
- **Event:** Each possible outcome of a variable is called an event.
- **Sample space:** The collection of all possible events is called sample space.
- **Random variables:** Random variables are used to represent the events and objects in the real world.
- **Prior probability:** The prior probability of an event is probability computed before observing new information.
- **Posterior Probability:** The probability that is calculated after all evidence or information has taken into account. It is a combination of prior probability and new information.

Probabilistic Reasoning

- **Conditional probability:**
- Conditional probability is a probability of occurring an event when another event has already happened.
- Let's suppose, we want to calculate the event A when event B has already occurred, "the probability of A under the conditions of B", it can be written as: $$P(A|B) = \frac{P(A \wedge B)}{P(B)}$$

Where

P($A \wedge B$)= Joint probability of A and B

P(B)= Marginal probability of B.

- If the probability of A is given and we need to find the probability of B, then it will be given as:

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

Probabilistic Reasoning

- **Example:**
- In a class, there are 70% of the students who like English and 40% of the students who likes English and mathematics, and then what is the percent of students those who like mathematics?

 Solution:

- Let, A is an event that a student likes Mathematics
- B is an event that a student likes English.

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

- **Hence, 57% are the students who like Mathematics.**

Bayes' Theorem

- Bayes' theorem is also known as **Bayes' rule, Bayes' law**, or **Bayesian reasoning**, which determines the probability of an event with uncertain knowledge.
- In probability theory, it relates the conditional probability and marginal probabilities of two random events.
- Bayes' theorem was named after the British mathematician **Thomas Bayes**. The **Bayesian inference** is an application of Bayes' theorem, which is fundamental to Bayesian statistics.
- It is a way to calculate the value of P(B|A) with the knowledge of P(A|B).
- Bayes' theorem allows updating the probability prediction of an event by observing new information of the real world.

Bayes' Theorem

- **Example**: If cancer corresponds to one's age then by using Bayes' theorem, we can determine the probability of cancer more accurately with the help of age.
- Bayes' theorem can be derived using product rule and conditional probability of event A with known event B:
- As from product rule we can write: $P(A \wedge B) = P(A|B) P(B)$
- Similarly, the probability of event B with known event A: $P(A \wedge B) = P(B|A) P(A)$
- Equating right hand side of both the equations, we will get:

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)} \qquad \text{(a)}$$

- The above equation (a) is called as **Bayes' rule** or **Bayes' theorem**. This equation is basic of most modern AI systems for **probabilistic inference**.

Bayes' Theorem

- It shows the simple relationship between joint and conditional probabilities. Here,
- P(A|B) is known as **posterior**, which we need to calculate, and it will be read as Probability of hypothesis A when we have occurred an evidence B.
- P(B|A) is called the likelihood, in which we consider that hypothesis is true, then we calculate the probability of evidence.
- P(A) is called the **prior probability**, probability of hypothesis before considering the evidence
- P(B) is called **marginal probability**, pure probability of an evidence.
- In the equation (a), in general, we can write P(B) = P(A)*P(B|Ai), hence the Bayes' rule can be written as:

$$P(A_i|B) = \frac{P(A_i)*P(B|A_i)}{\sum_{i=1}^{k} P(A_i)*P(B|A_i)}$$

- Where A_1, A_2, A_3,........, A_n is a set of mutually exclusive and exhaustive events.

Applying Bayes' rule

Example - 1:

Question: what is the probability that a patient has diseases meningitis with a stiff neck?

Given Data: A doctor is aware that disease meningitis causes a patient to have a stiff neck, and it occurs 80% of the time. He is also aware of some more facts, which are given as follows:

- The Known probability that a patient has meningitis disease is 1/30,000.
- The Known probability that a patient has a stiff neck is 2%.

Let **a** be the proposition that patient has stiff neck and **b** be the proposition that patient has meningitis. , so we can calculate the

$$P(b|a) = \frac{P(a|b)P(b)}{P(a)} = \frac{0.8*(\frac{1}{30000})}{0.02} = 0.001333333.$$

- $P(a|b) = 0.8$
- $P(b) = 1/30000$
- $P(a) = .02$

- Hence, we can assume that 1 patient out of 750 patients has meningitis disease with a stiff neck.

Applying Bayes' rule

Example-2:

Question: From a standard deck of playing cards, a single card is drawn. The probability that the card is king is 4/52, then calculate posterior probability P(King|Face), which means the drawn face card is a king card.

- **Solution:** $$P(king|face) = \frac{P(Face|king) * P(King)}{P(Face)} \quad ....(i)$$
- P(king): probability that the card is King= 4/52= 1/13
- P(face): probability that a card is a face card= 3/13
- P(Face|King): probability of face card when we assume it is a king = 1
- Putting all values in equation (i) we will get:

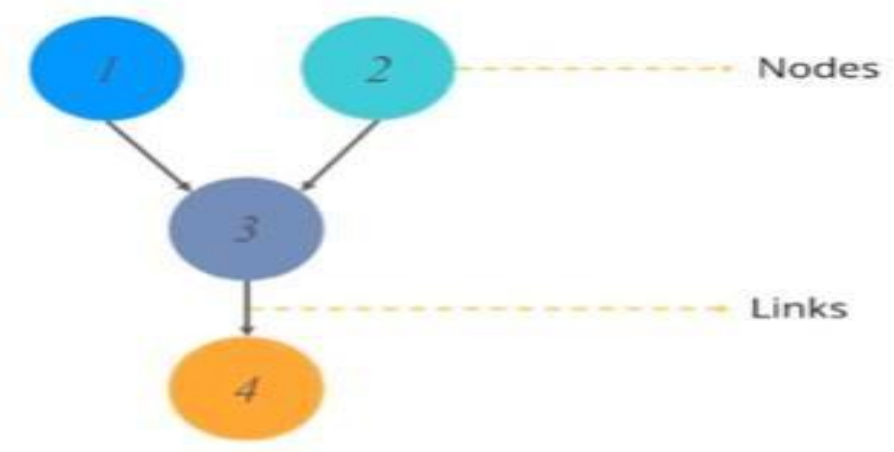
$$P(king|face) = \frac{1 * (\frac{1}{13})}{(\frac{3}{13})} = \mathbf{1/3}, \text{ it is a probability that a face card is a king card.}$$

Belief Networks

- A belief network (Bayes net) represents the dependence between variables.
- Components of a belief network graph:
  - Nodes
    - These represent variables
  - Links
    - X points to Y if X has a direct influence on Y
  - Conditional probability tables
    - Each node has a CPT that quantifies the effects the parents have on the node
  - The graph has no directed cycles

Bayesian Network

- Bayesian network falls under the category of probabilistic Graphical Modeling(PGM) technique that is used to compute uncertainties by using the concept of probability.
- Bayesian network is known as **belief network** or **casual network**
- Bayesian network is based on the concept of probability. It can be represented by using a directed acyclic graph.
- Directed acyclic graph is used to represent Bayesian network and like any other statistical group. DAG contains a set of nodes and links, where the link denote the relationship between the nodes.



Bayesian Network

- DAG models the uncertainty of an event occurring based on conditional probability distribution of each random variable.
- Conditional probability table is used to represent this distribution of each variable in the network.

Joint Probability is a measure of two events happening at the same time, i.e., p(A and B). The probability of the intersection of A and B may be written $p(A \cap B)$.

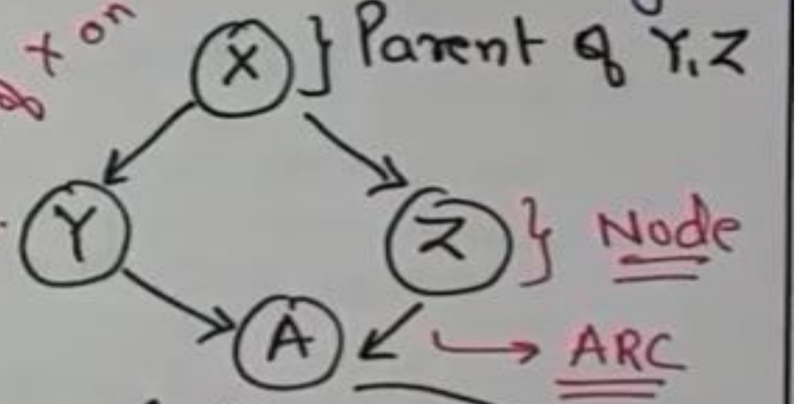
Conditional Probability of an event B is the probability that the event will occur given that an event A has already occurred.

$p(B|A)$: probability of event B occurring, given that event A occurs.
If A and B are **dependent** events : $P(B|A) = P(A \text{ and } B) / P(A)$
If A and B are **independent** events: $P(B|A) = P(B)$

Bayesian Belief Network

Bayesian Belief N/w in AI: It defines probabilistic independencies and dependencies among the Variables in the N/w.

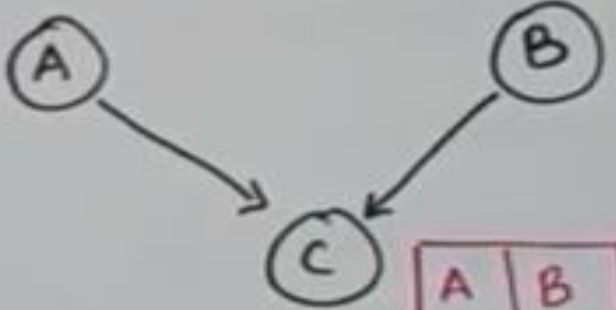
- "It is a probabilistic graphical model which represents a set of Variables and their Conditional dependencies using a directed acyclic graph". (DAG).
- Built from Probability distribution.
- Consists of
  1. DAG.
  2. Table of Conditional Probabilities.
- Node: Corresponds to a Random Variable → Continuous / → Discrete
- Arc/Directed arrows: rep. Casual relationship or Conditional probabilities among random Variables.



Bayesian Belief Network

To propagate belief in Bayesian N/W, initial "Directed Acyclic Graph" is converted into an undirected graph in which the arcs can be used to transmit probabilities in dirⁿ of evidence.

T	.002
F	.998

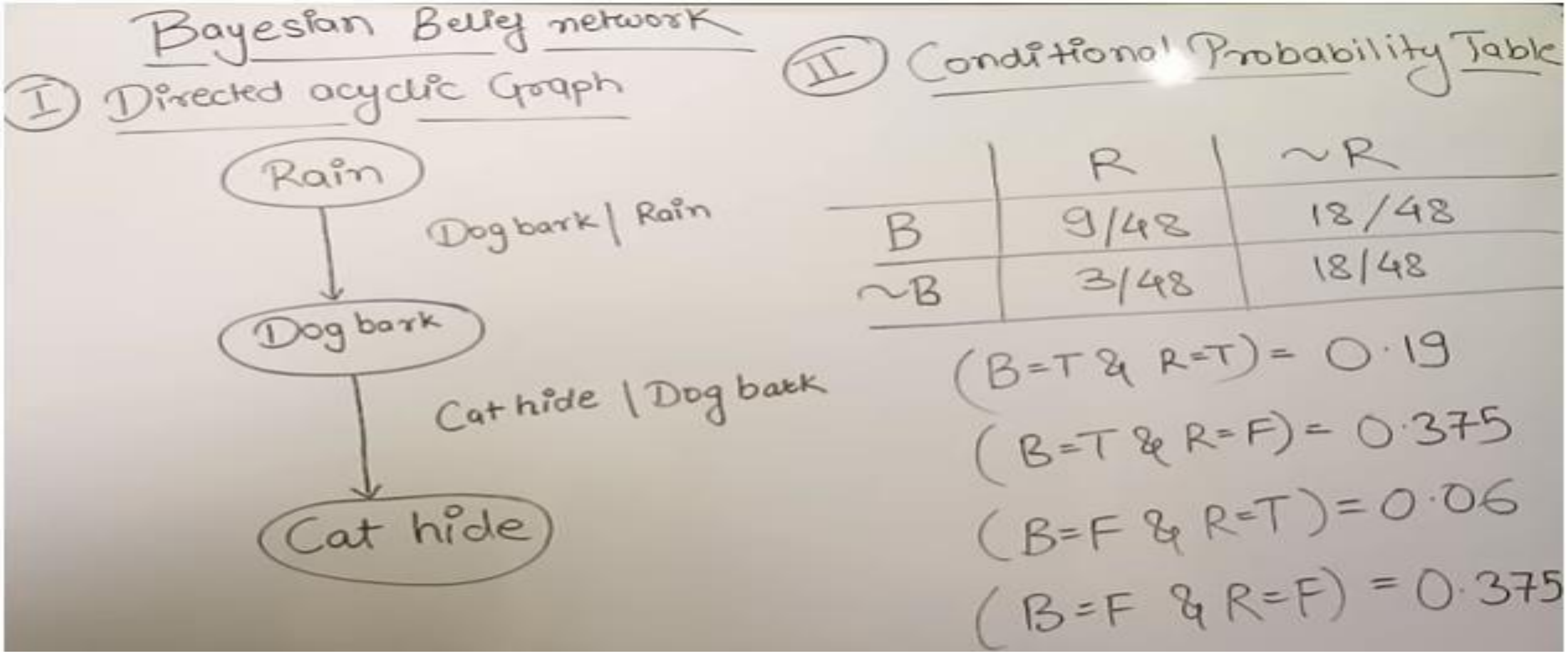


T	0.001
F	.999

A	B	P(C=T)	C=F
T	T	–	–
T	F	–	–
F	T	–	–
F	F	–	–

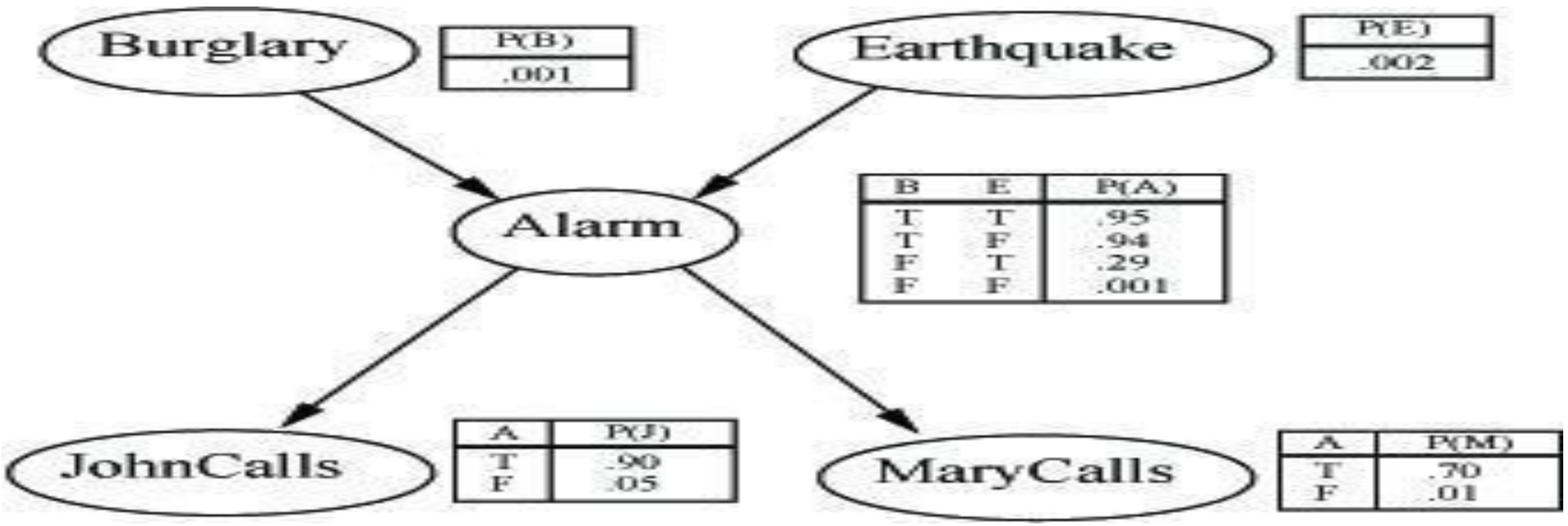
Bayesian Network Example

- Convenient for representing probabilistic relation between multiple events.



Bayesian Network Example2

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

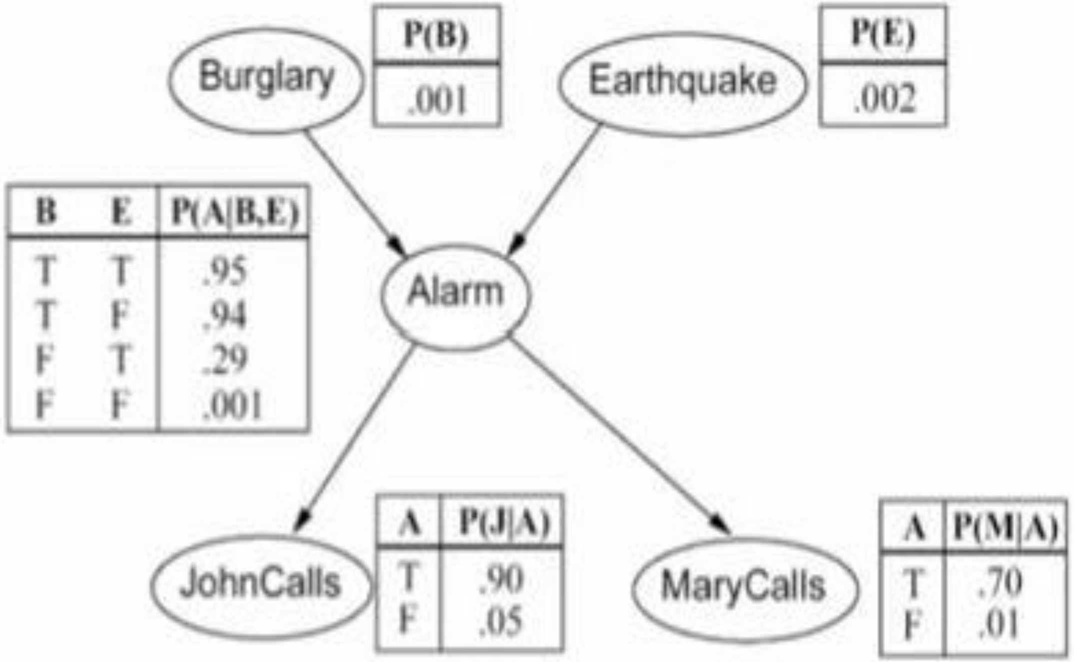


Bayesian Network Example2

- You have a new burglar alarm installed at home.
- It is fairly reliable at detecting burglary, but also sometimes responds to minor earthquakes.
- You have two neighbors, John and Merry , who promised to call you at work when they hear the alarm.
- John always calls when he hears the alarm, but sometimes confuses telephone ringing with the alarm and calls too.
- Merry likes loud music and sometimes misses the alarm.
- Given the evidence of who has or has not called, we would like to estimate the probability of a burglary.

Bayesian Network Example2

What is the probability that the alarm has sounded but neither a burglary nor an earthquake has occurred, and both John and Merry call?



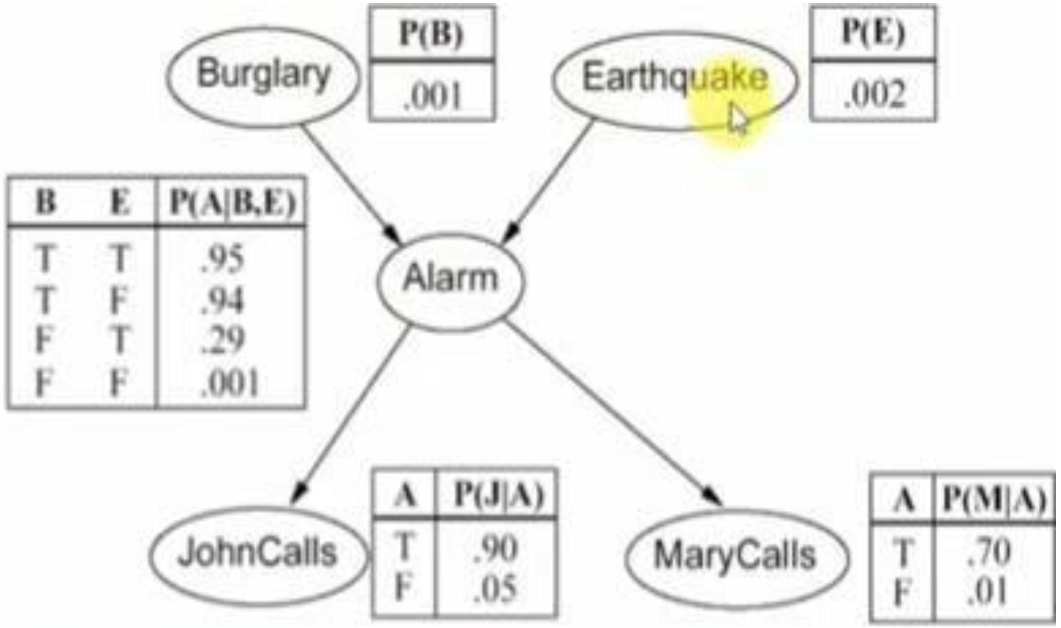
$$P(j \wedge m \wedge a \wedge \neg b \wedge \neg e) = P(j \mid a)\, P(m \mid a)\, P(a \mid \neg b, \neg e)\, P(\neg b)\, P(\neg e)$$

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

$$= 0.00062$$

Bayesian Network Example2

2. What is the probability that John call?



Solution:

$$P(j) = P(j \mid a)\, P(a) + P(j \mid \neg a)\, P(\neg a)$$

$$= P(j|a)\{P(a|b,e)*P(b,e)+P(a|\neg b,e)*P(\neg b,e)+P(a|b,\neg e)*P(b,\neg e)+P(a|\neg b,\neg e)*P(\neg b,\neg e)\}$$

$$+ P(j|\neg a)\{P(\neg a|b,e)*P(b,e)+P(\neg a|\neg b,e)* P(\neg b,e)+P(\neg a|b,\neg e)* P(b,\neg e)+P(\neg a|\neg b, \neg e)* P(\neg b, \neg e)\}$$

$$= 0.90 * 0.00252 + 0.05 * 0.9974 = 0.0521$$

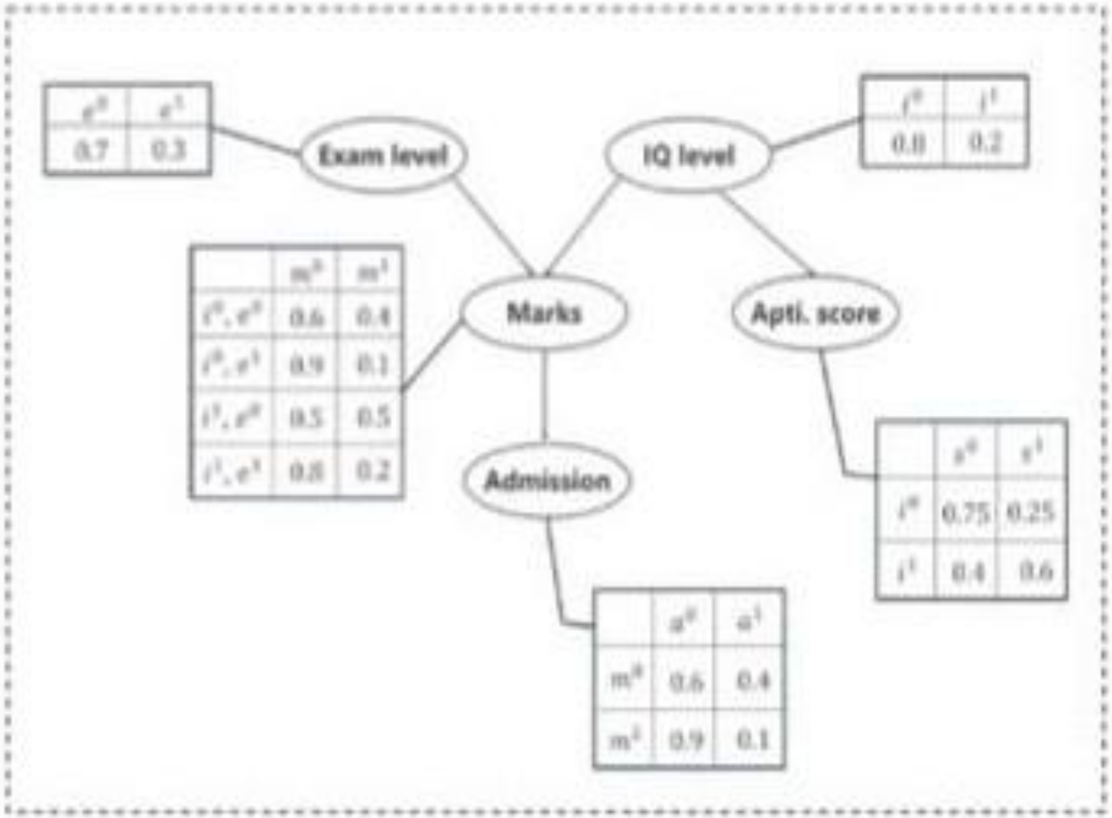
Bayesian Network Example

- There should not be a cycle.

Create a Bayesian Network that will model the marks (m) of a student on his examination.

The marks will depend on:

- ***Exam level** (e): (difficult, easy)*
- ***IQ** of the student (i): (high, low)*
- *Marks -> **admitted (a)** to a university*
- *The IQ -> **aptitude score (s)** of the student*

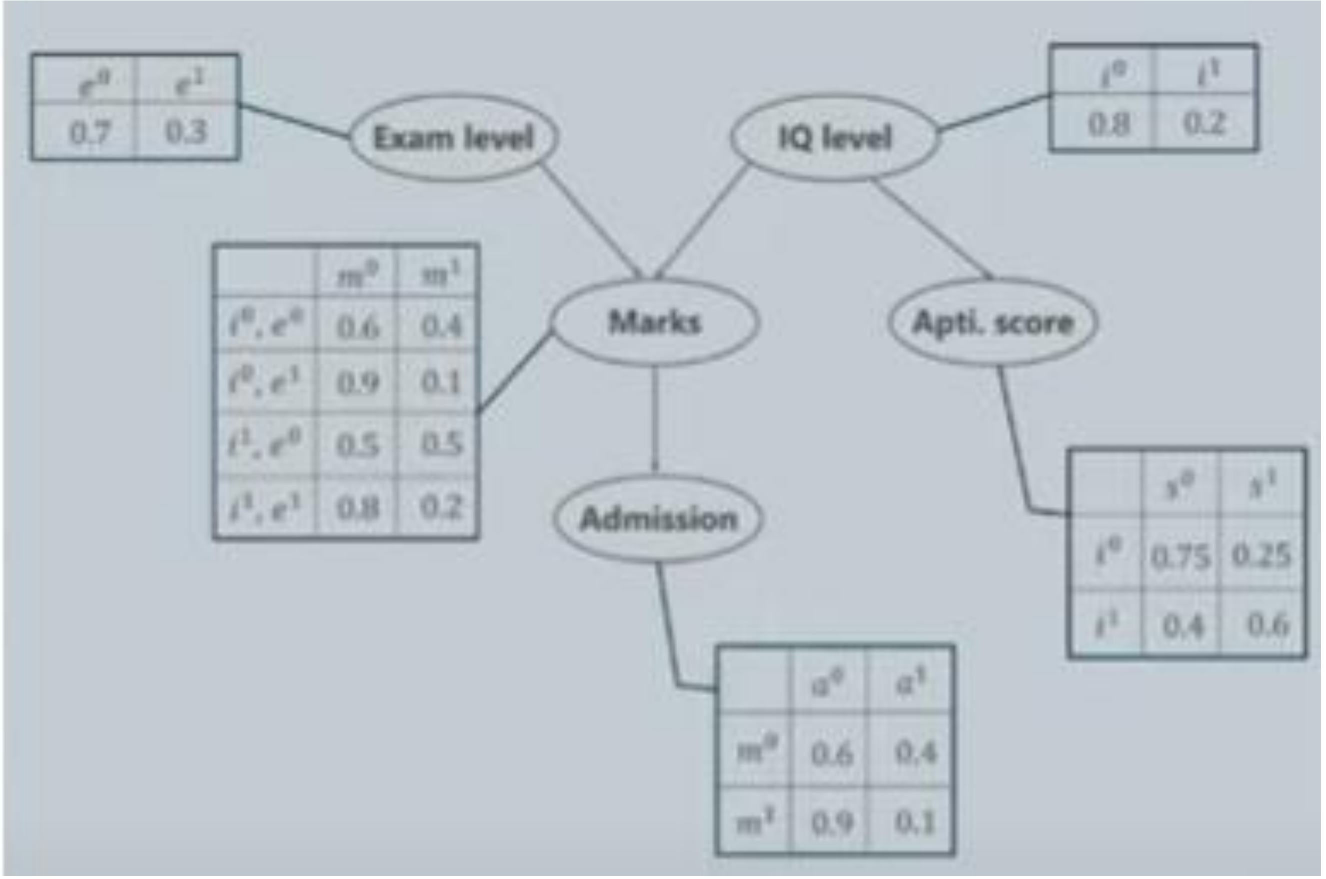


Bayesian Network Example

Factorizing Joint Probability Distribution:

p(a, m, i, e, s) = p(a | m) p(m | i, e) p(i) p(e) p(s | i)

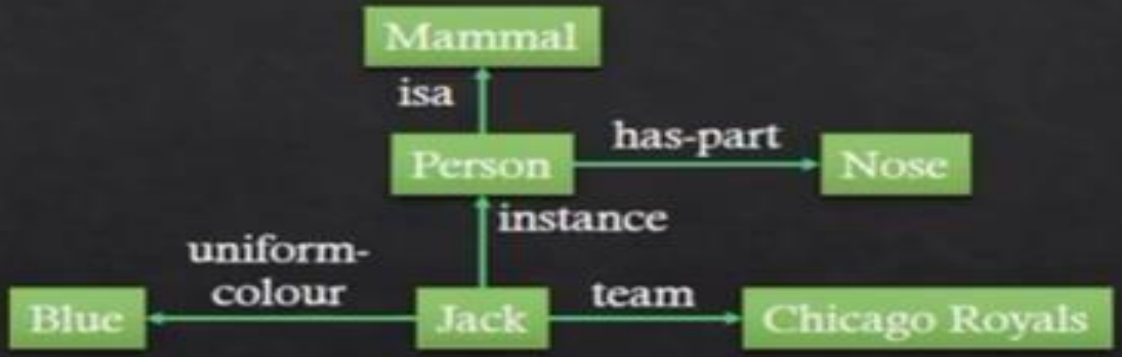
- p(a | m): CP of student admit -> marks
- p(m | i, d): CP of the student's marks -> (IQ & exam level)
- p(i): Probability -> IQ level
- p(d): Probability -> exam level
- p(s | i) CP of aptitude scores ->IQ level



Semantic Nets

- Semantic Nets – way of representing knowledge and are based on property inheritance.
- How to represent English statements using semantic nets?
- Represent knowledge with the help of property inheritance we take the help of structures called slot and filler structures.

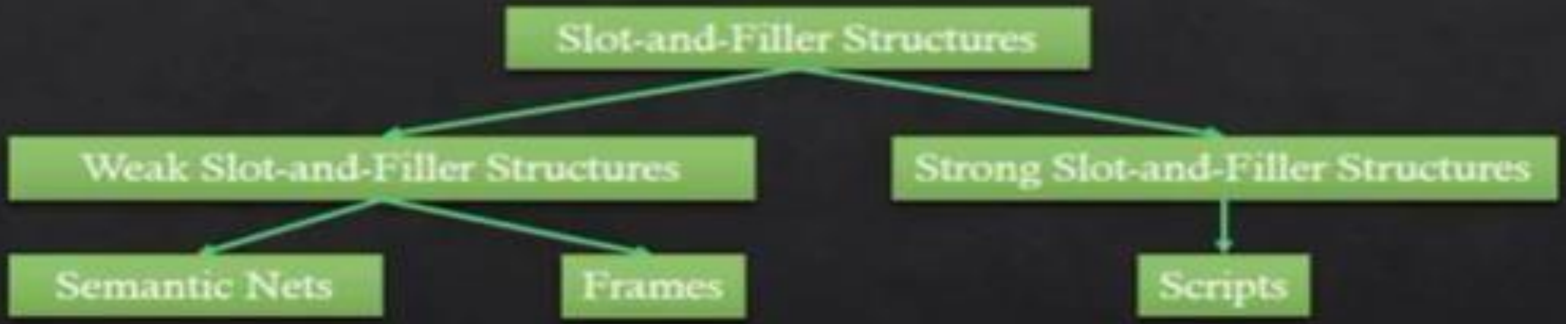
- Slot and filler structures are the devices to support Property Inheritance using isa and instance links.



- <u>Arcs</u>: Represent relationships among the nodes
- <u>Nodes</u>: classes, or objects, or values of attributes of an object.
- A slot is an attribute-value pair and a filler is the actual value that the slot can take.

Semantic Nets

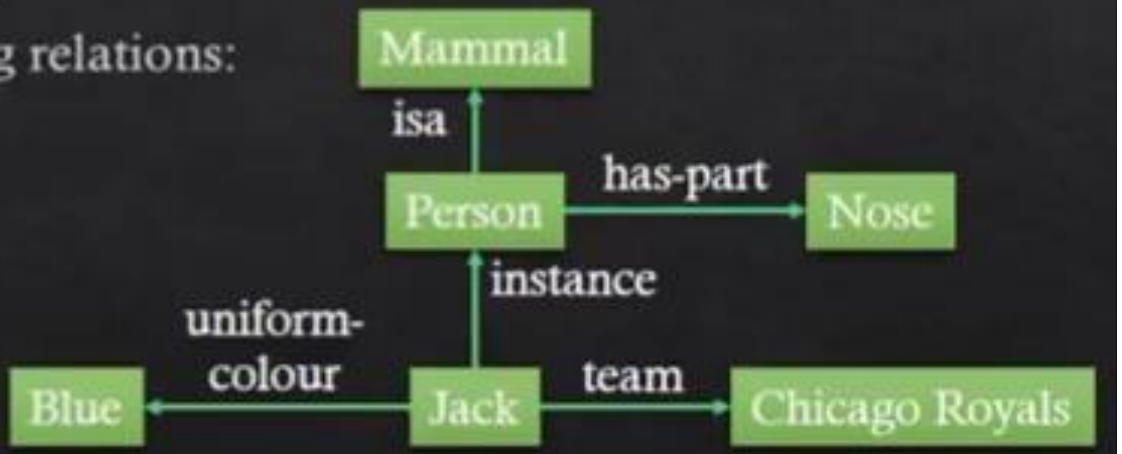
- Advantages of Slot-and-Filler structures
  - They support inheritance,
  - They support monotonic as well as non-monotonic reasoning effectively.
  - They make it easy to describe properties of relations.
  - Since they follow Object-Oriented approach, there is modularity



- Slot and filler structures are of two types:
  - Weak Slot-and-Filler Structures: Very little importance is given to the specific knowledge the structure should contain
  - Strong Slot-and-Filler structures: Specific commitments are made to the context of the representation.

Semantic Nets

- Semantics: Meaning of words
- Net: Network.
- In Semantic Nets, the meaning of a concept comes from the ways in which it is connected to other concepts.
- Information is represented as a set of nodes connected to each other by a set of labelled arcs, which represent relationships among nodes.
- In our example, we have represented the following relations:
  1. isa(Person, Mammal)
  2. has-part(Person, Nose)
  3. instance(Jack, Person)
  4. team(Jack, Chicago-Royals)
  5. uniform-color(Jack, Blue)

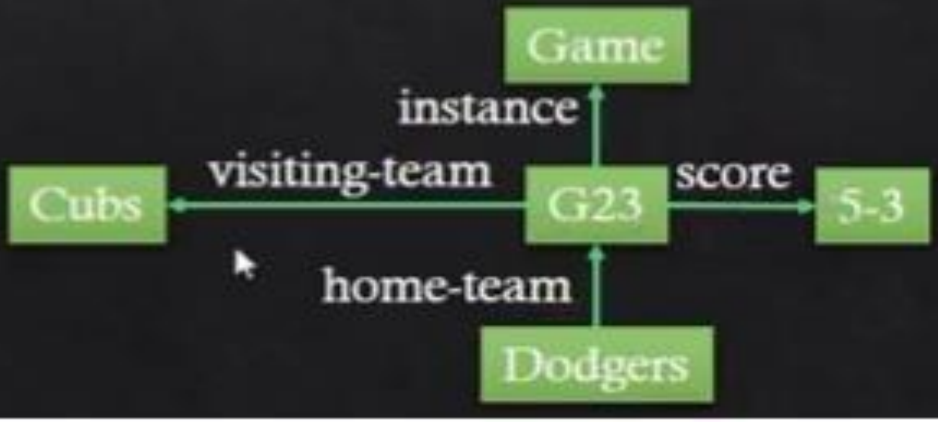


- But, from the same network, we could use inheritance to derive the additional relation:
  6. has-part(Jack, Nose)

Representing N-place Predicates using Semantic Nets

- <u>Single Place Predicate:</u>
  - Marcus is a man
    - man(Marcus)
    - instance(Marcus, Man)

- <u>N-Place Predicate:</u>
  - score(Cubs, Dodgers, 5-3)
    - three place predicate
      - visiting team, Cubs
      - home team, Dodgers
      - Score
    - Represent the game as a base class
    - G23 as instance of Game, with attributes:
      - Visiting team
      - Home team
      - Score
    - Game can be a class, instance can be G23



Representing Declarative Statement using Semantic Nets

- A declarative statement is a statement that states a fact.
  - Prabha likes riding a bike.
  - Vikram exchanges good books.
  - Pratima sang a classical song.
- Consider the statement: John gave the book to Mary
- Organise, the information into base class, derived class, their attributes, the values of those attributes, objects, their attributes and their values.

Representing Declarative Statement using Semantic Nets

- Consider the statement: John gave the book to Mary
- The event is an instance of the act of giving.
  - Give: class node
  - EV1: object of class give
  - John is the doer or the agent of the instance of giving.
  - The receiver of the instance of giving is Mary: beneficiary. The value of beneficiary attribute is Mary.
  - the item given is an instance of a class of books. That means we'll have to create a class node called book and its instance called say BK23 and this instance is the filler of the attribute item

Frame Representation

- When an **agent encounters a new situation**, it will need to **retrieve information to act rationally** in that situation. This information is likely to be **multi-faceted and hierarchical**. One way of **structuring the knowledge is in terms of frames**. These are **frameworks consisting of slots**.
- Each **slot containing information in various representations** like **logical sentences, production rule, frame** etc. Each framework represents a stereotypical object or situation.
- Whenever an **agent encounters an object or situation** which fits the stereotype, **agent retrieve the framework and change some default information or fill the blank information**. Some of the information is procedural, so that when a blank is filled in with certain values, a procedure must be carried out. In this way frame dictate how to act rationally in situations.
- **Types of Information store in slot**
    - Information for choosing the frame
    - Information about relationships between frames.
    - Procedures to be carried out.
    - Default information.
    - Blank slots.

Frame Representation

- Information for choosing the frame
  - It may also be **information about situations or descriptors** for the stereotype the frame represents. **E.g. name ,id**
- Information about relationships between frames
  - **Two frames should never be considered at the same time whether this frame is a generalisation or specialisation of another frame.**
- Procedures to be carried out
  - It is **rational action** an agent should do in a situation **where a particular value for a slot has been identified.**
- Default information
  - It's the values when **certain information** required for the frame is **missing**.
  - Default information is used in **choosing actions until more specific information is found**.
- Blank slots
  - These are flagged to be **left blank unless required for a particular task**.

Frames

- It means of **representing common sense knowledge**. Knowledge is **organized into small packets called “Frames”.** All frames of a given situation constitute the system.
- A frame can be defined as **a structure that has slots for various objects & a collection of frames consist of expectation for a given situation**.
- Frame are used to **represent two types of knowledge** viz. **declarative/factual and procedural**, declarative & procedural Frames.
- A frame that merely contains description about objects is call a declarative type situational frame.

Name : Computer Centre	
A/c	Stationary cupboard
Computer	Dumb terminals
Printer	

Name of the frame

Slots in the frame

Frames

- **Frames which have procedural knowledge embedded in it are called action procedure frames.**

  The action frame has the following slots:

  - **Actor slot** which holds information @ **who** is performing the activity.
  - **Source Slot** hold information from **where** the action has to begin.
  - **Destination slot** holds information about the **place** where **action has to end.**
  - **Task slot,** this **generates the necessary sub frames required to perform the operation**.

<table>
<tr><td colspan="3">Name : Cleaning the ict of carburetor</td></tr>
<tr><td colspan="3">Actor</td></tr>
<tr><td colspan="3">Expert
Object
Source Destination</td></tr>
<tr><td>Scooter</td><td colspan="2">Scooter</td></tr>
<tr><td colspan="3">Task 1 Task 2
Task 3</td></tr>
<tr><td>Remove Carburetor</td><td>Clean Nozzle</td><td>Fix Carburetor</td></tr>
</table>

Fuzzy Logic

- Fuzzy Logic looks at the world in imprecise terms, in much **the same way that our brain takes in information** (e.g. temperature is hot, speed is slow), **then responds with precise actions**.
- **Fuzzy Logic is an approach to computing based on "degrees of truth" rather than the usual "true or false" logic**. The human brain can reason with uncertainties, vagueness, and judgments. Computers can only manipulate precise valuations. Fuzzy logic is an attempt to combine the two techniques.
- Fuzzy logic is in fact, **a precise problem-solving methodology**. Fuzzy logic differs from classical logic in that statements are no longer black or white, true or false, on or off.
- **In traditional logic an object takes on a value of either zero or one. In fuzzy logic, a statement can assume any real value between 0 and 1, representing the degree to which an element belongs to a given set.** Works with imprecise statements such as: In a process control situation, **"*If* the temperature is moderate and the pressure is high, *then* turn the knob slightly right"**
- **It represent uncertainty**
- **Represent with degree**
- **Represent the belongingness of a member of a crisp set to fuzzy set.**

Fuzzy Logic Applications

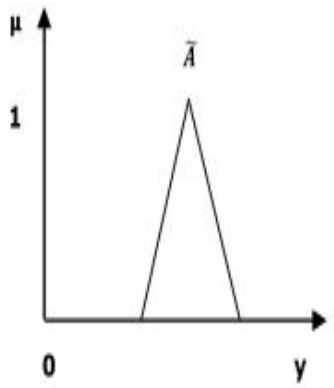
1. For **washing machines**, **Fuzzy Logic control is almost becoming a standard feature**: fuzzy controllers to load-weight, fabric-mix, and dirt sensors and automatically set the wash cycle for the best use of power, water, and detergent.
2. **NASA** has studied fuzzy control for **automated space docking**: simulations show that a fuzzy control system can **greatly reduce fuel consumption**
3. **Canon** developed an **auto-focusing camera** that uses a charge-coupled device (CCD) to measure the clarity of the image in six regions of its field of view and use the information provided to determine if the image is in focus. It also tracks the rate of change of lens movement during focusing, and controls its speed to prevent overshoot.
4. **Nissan – fuzzy automatic transmission, fuzzy anti-skid braking system**
5. **CSK, Hitachi – Hand-writing Recognition**
6. **Ricoh, Hitachi – Voice recognition**
7. **Automotive system for speed control, traffic control.**

Fuzzy Logic-Set Theory

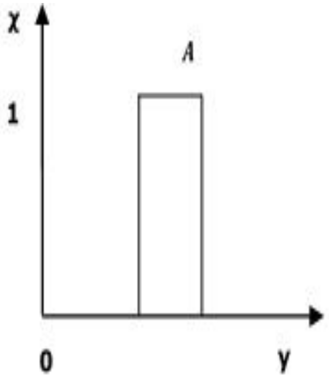
- The **Classical/Crisp set** is defined in such a way that the **universe of discourse is spitted into two groups members and non-members**. Hence, In case classical sets, **no partial membership exists**.
- In a Crisp set, membership or non-membership of element "x" in set A is described by a characteri$\mu_A(x)$, where $\mu_A(x) = 1$ if $x \in A$ and $\mu_A(x) = 0$ if $x \notin A$.
- **Fuzzy set** is a set having **degrees of membership** between 1 and 0. Fuzzy sets are represented with **tilde character(~).** A fuzzy set A on a universe of discourse U is characterized by a membership function $\mu_{\tilde{A}}(y)$ that takes values in the interval [0, 1].
- **Classical set** contains elements that satisfy **precise properties** of membership while **fuzzy set** contains elements that satisfy **imprecise properties** of membership.
- Fuzzy sets allows partial membership which means that it **contain elements that have varying degrees of membership in the set**. E.g. **{hot, warm, cool, cold}, { not cold, barely cold, bit cold, not quit cold, cold}**
- A **fuzzy set** is a mapping of a **set** of real numbers (x_i) onto membership values (u_i) that (generally) lie in the range [0, 1]. In this **fuzzy** package a **fuzzy set** is **represented** by a **set** of pairs u_i/x_i, where u_i is the membership value for the real number x_i. We can **represent** the **set** of values as $\{ u_1/x_1 \; u_2/x_2 \ldots u_n/x_n \}$.

Fuzzy Logic-Set Theory

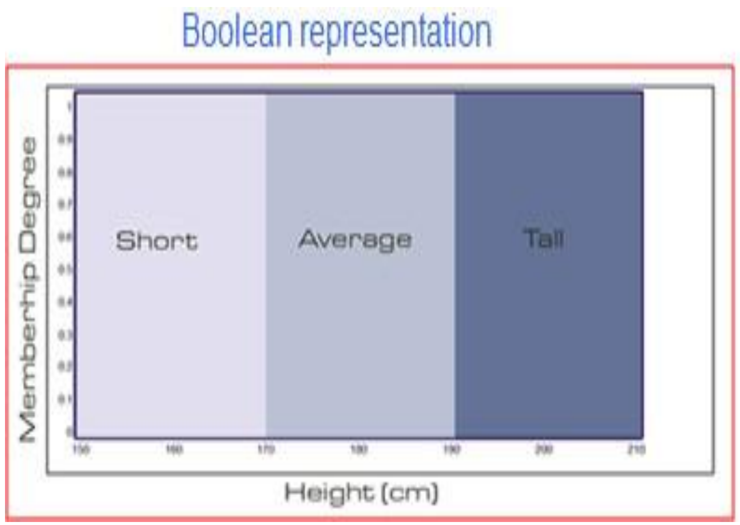
- From this, we can understand the difference between classical set and fuzzy set. **Classical set contains elements that satisfy precise properties of membership while fuzzy set contains elements that satisfy imprecise properties of membership.**

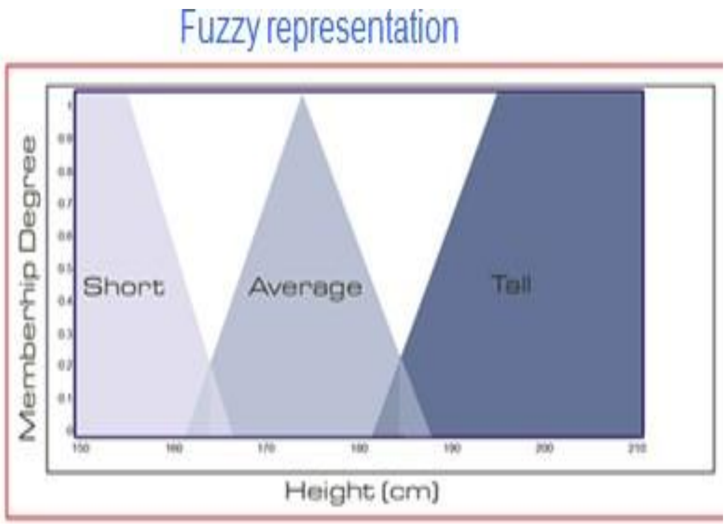


Membership Function of Fuzzy set $\tilde{A}$



Membership Function of classical set A





Example: Is a man whose height is 5‟ 11 average or tall?

- A fuzzy system might say that he is partly medium and partly tall.
- In fuzzy terms, the height of the man would be classified within a range of [0,1] as average to a degree of 0.6, and tall to a degree of 0.4.

Fuzzy Logic-Set Theory

- In other words, **FL recognizes not only clear-cut, black-and-white alternatives, but also the infinite gradations in between**. **Fuzzy reasoning eliminates the vagueness by assigning specific numbers to those gradations.** These **numeric values are then used to derive exact solutions to problems**.
- A fuzzy set $\widetilde{A}$ in the universe of information U can be defined as a set of ordered pairs and it can be represented mathematically as − $\widetilde{A} = \{(y, \mu_{\widetilde{A}}(y)) \mid y \in U\}$

Here $\mu_{\widetilde{A}}(y)$ree of membership of y , assumes values in the range from 0 to 1, i.e., $\mu_{\widetilde{A}}(y) \in [0, 1]$

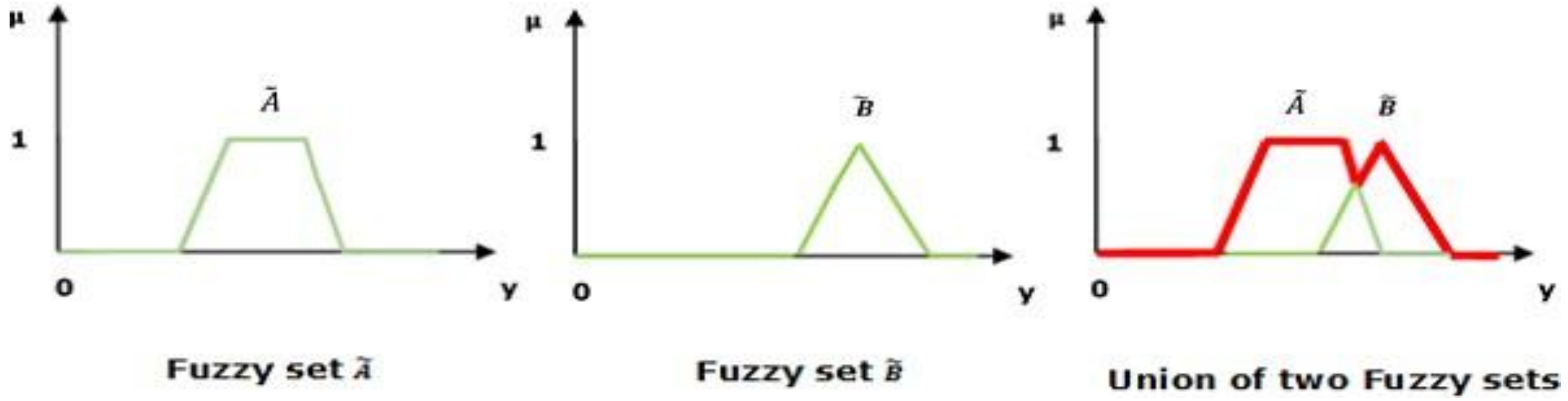
OR

$$\widetilde{A} = \left\{ \frac{\mu_{\widetilde{A}}(y_1)}{y_1} + \frac{\mu_{\widetilde{A}}(y_2)}{y_2} + \frac{\mu_{\widetilde{A}}(y_3)}{y_3} + \ldots \right\}$$

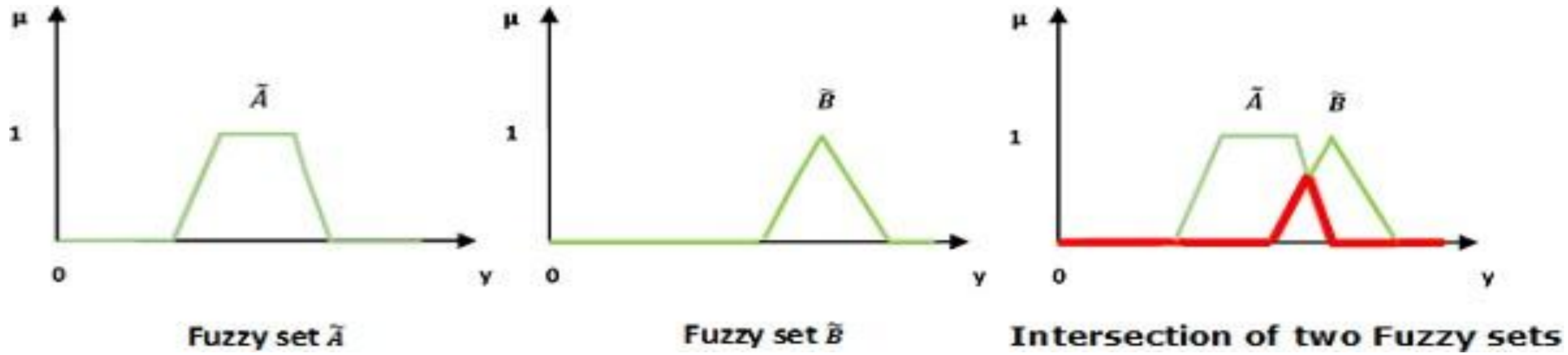
$$= \left\{ \sum_{i=1}^{n} \frac{\mu_{\widetilde{A}}(y_i)}{y_i} \right\}$$

Operations of Fuzzy Set

- Union/Fuzzy “OR” $\mu_{A \cup B}(x) = \max\left[\mu_A(x), \mu_B(x)\right]$

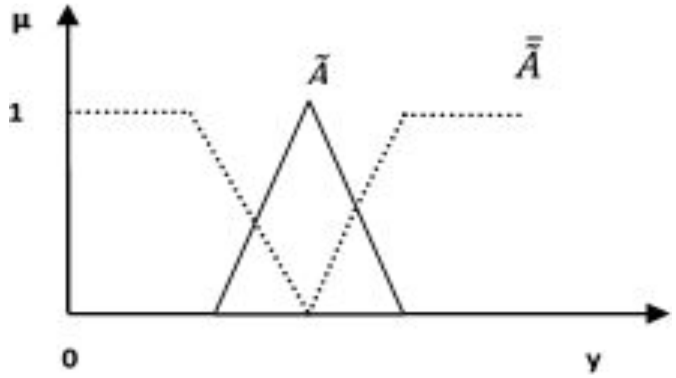


- Intersection/Fuzzy "AND“ $\mu_{A \cap B}(x) = \min\left[\mu_A(x), \mu_B(x)\right]$



Operations of Fuzzy Set

- Complement/Fuzzy "NOT" $\mu_{\bar{A}}(x) = 1 - \mu_A(x)$



Properties of Fuzzy Set

Commutative Property:

Having two fuzzy sets $\tilde{A}$ and $\tilde{B}$, this property states –

$$\tilde{A} \cup \tilde{B} = \tilde{B} \cup \tilde{A}$$

$$\tilde{A} \cap \tilde{B} = \tilde{B} \cap \tilde{A}$$

Identity Property:

For fuzzy set $\tilde{A}$ and universal set U, this property states –

$$\tilde{A} \cup \varphi = \tilde{A}$$

$$\tilde{A} \cap U = \tilde{A}$$

$$\tilde{A} \cap \varphi = \varphi$$

$$\tilde{A} \cup U = U$$

De Morgan‟s Law:

This law plays a crucial role in proving tautologies and contradiction. This law states –

$$\overline{\tilde{A} \cap \tilde{B}} = \overline{\tilde{A}} \cup \overline{\tilde{B}}$$

$$\overline{\tilde{A} \cup \tilde{B}} = \overline{\tilde{A}} \cap \overline{\tilde{B}}$$

Distributive Property:

Having three fuzzy sets $\tilde{A}$, $\tilde{B}$ and $\tilde{C}$, this property states –

$$\tilde{A} \cup \left(\tilde{B} \cap \tilde{C}\right) = \left(\tilde{A} \cup \tilde{B}\right) \cap \left(\tilde{A} \cup \tilde{C}\right)$$

$$\tilde{A} \cap \left(\tilde{B} \cup \tilde{C}\right) = \left(\tilde{A} \cap \tilde{B}\right) \cup \left(\tilde{A} \cap \tilde{C}\right)$$

Transitive Property:

Having three fuzzy sets $\tilde{A}$, $\tilde{B}$ and $\tilde{C}$, this property states –

$$If\, \tilde{A} \subseteq \tilde{B} \subseteq \tilde{C},\, then\, \tilde{A} \subseteq \tilde{C}$$

Idempotency Property:

For any fuzzy set $\tilde{A}$, this property states –

$$\tilde{A} \cup \tilde{A} = \tilde{A}$$

$$\tilde{A} \cap \tilde{A} = \tilde{A}$$

Involution Property:

For any fuzzy set $\tilde{A}$, this property states –

$$\overline{\overline{\tilde{A}}} = \tilde{A}$$

Fuzzy Logic Rule Base

- It contains the **set of rules and the IF-THEN conditions** provided by the experts **to govern the decision making system,** on the basis of linguistic information.
- Recent developments in fuzzy theory offer several effective methods for the **design and tuning of fuzzy controllers.** Most of these developments reduce the number of fuzzy rules.

Examples:

R1: <u>if</u> temperature is high <u>then</u> Climate is hot

R2: <u>if</u> outlook is sunny and <u>if</u> humidity is high <u>then</u> Decision is <u>No</u>.

R3: <u>if</u> outlook is sunny and <u>if</u> humidity is low <u>then</u> Decision is <u>Yes</u>.

R4: <u>if</u> Tree_distance is SOMEWHAT close AND Tree_angle is small_positive <u>then</u> Turn slightly left

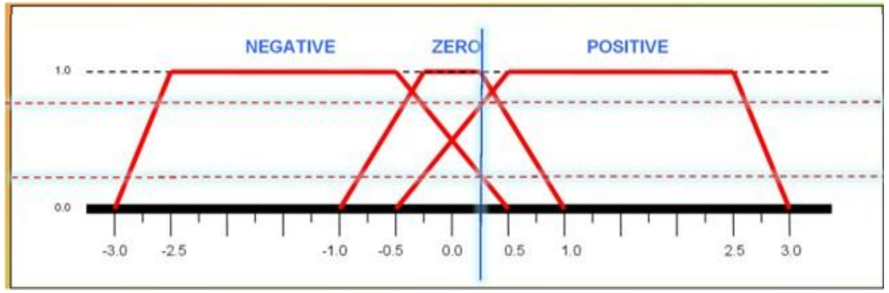
Fuzzification and Defuzzification

Fuzzification:- Input variables are assigned degrees of membership in various classes. The purpose of fuzzification is **to map the inputs from a set of sensors** (or features of those sensors such as amplitude or spectrum) **to values from 0 to 1 using a set of input membership functions**.

Example: Fuzzy Sets = { Negative, Zero, Positive }

Assuming that we are using trapezoidal membership functions.

Crisp Input: x = 0.25



Defuzzification:- It is used **to convert the fuzzy sets obtained by inference engine into a crisp value**. There are several defuzzification methods available and the best suited one is used with a specific expert system to reduce the error.

Different methods of defuzzification are listed below:

- Maxima Methods
- Centroid Method
- Weighted Average Method
- Mean-Max Membership
- Smallest of Maxima and Largest of Maxima
- Center of Maxima

Fuzzy Inference System

- A fuzzy inference system **(FIS)** is a system that **uses fuzzy set theory to map inputs** (*features* in the case of fuzzy classification) **to outputs** (*classes* in the case of fuzzy classification). This is a method to map an input to an output using fuzzy logic. **Based on this mapping process, the system takes decisions and distinguishes patterns.**
- **Characteristics** of Fuzzy Inference System
  - The output from FIS is always a fuzzy set irrespective of its input which can be fuzzy or crisp.
  - It is necessary to have fuzzy output when it is used as a controller.
  - A defuzzification unit would be there with FIS to convert fuzzy variables into crisp variables.
- **Functional Blocks of FIS**
  - **Rule Base** − It contains fuzzy IF-THEN rules.
  - **Database** − It defines the membership functions of fuzzy sets used in fuzzy rules.
  - **Decision-making Unit** − It performs operation on rules.
  - **Fuzzification Interface Unit** − It converts the crisp quantities into fuzzy quantities.
  - **Defuzzification Interface Unit** − It converts the fuzzy quantities into crisp quantities.

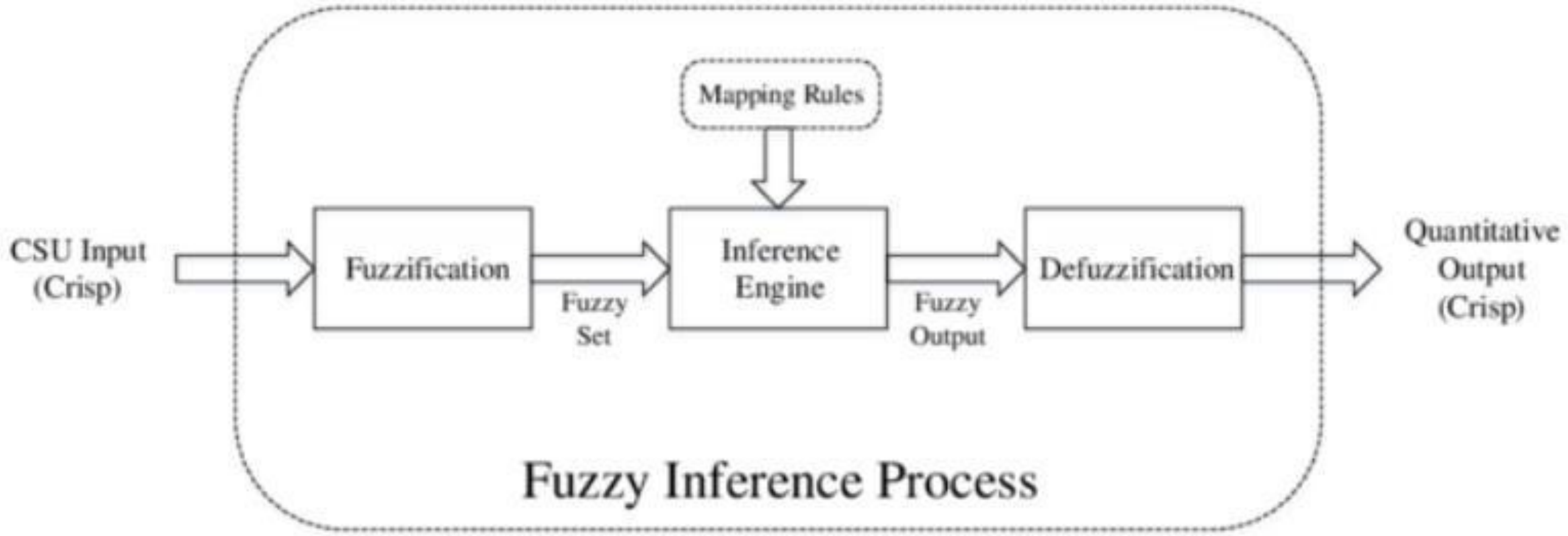
Fuzzy Inference System

- **Fuzzy Logic Rule Base**: **Fuzzy rule-based** systems are **rule-based** systems, where **fuzzy** sets and **fuzzy logic** are used as tools for representing different forms of knowledge about the problem at hand, as well as for modeling the interactions and relationships existing between its variables. E.g.

  R1: IF temperature is hot AND humidity is high. THEN fan speed is fast.

  The degree of truth assigned to temperature is hot and to humidity is high.
- **Fuzzification:** It is used to convert inputs i.e. crisp numbers into fuzzy sets. Crisp inputs are basically the exact inputs measured by sensors and passed into the control system for processing, such as temperature, pressure, etc.
- **Inference Engine:** It determines the matching degree of the current fuzzy input with respect to each rule and decides which rules are to be fired according to the input field. Next, the fired rules are combined to form the control actions.
- **Defuzzification:** It is the process of producing a quantifiable result in Crisp logic, given fuzzy sets and corresponding membership degrees. It is the process that maps a fuzzy set to a crisp set.

Fuzzy Inference System Architecture

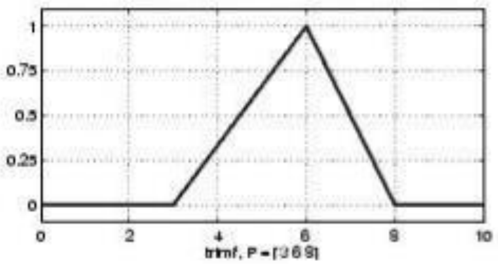


- The working of the FIS consists of the following steps −
- A **fuzzification** unit converts the crisp input into fuzzy input.
- A **knowledge base** - collection of rule base and database is formed upon the conversion of crisp input into fuzzy input.
- The **defuzzification** unit fuzzy input is finally converted into crisp output.

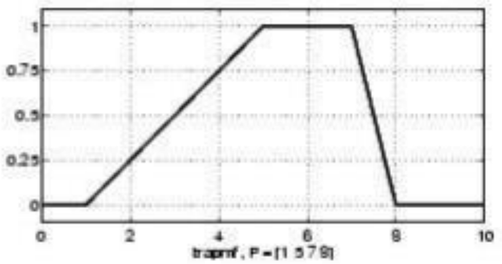
Fuzzy Membership Functions

4 common types of fuzzy membership functions:

- triangular (3 parameters) trapezoidal(4 para)

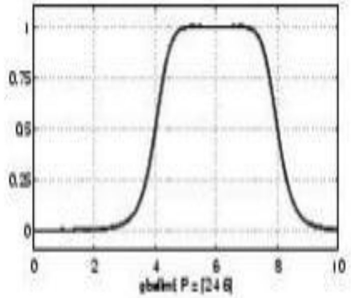


trimf



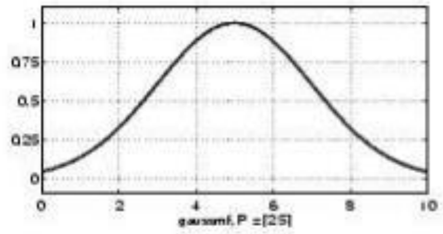
trapmf

- generalised bell (3 parameters)

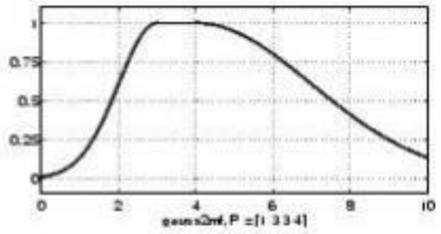


gbellmf

- gaussian (2 parameters)



gaussmf



gauss2mf

Defuzzification Methods

- **Max membership principle**: In this method, the defuzzifier examines the aggregated fuzzy set and chooses that output *y for which* $\mu_B(y)$*he maximum*
- **Centroid method:** centre of mall, centre of gravity or area. In this method, the defuzzifier determines the center of gravity (centroid) of B and uses that value y_i' as the output of the FLS. For a continuous aggregated fuzzy set, the centroid is given by

$$y' = \frac{\int_s y_i \mu_B(y)\,dy}{\int_s \mu_B(y)\,dy}$$

- **Weighted average method:** Valid for symmetrical output membership function.

Each membership function is weighted by its max membership value.

$$X^* = \frac{\sum Mc(\bar{xi}).\bar{xp}}{\sum MC(\bar{xi})}$$

$\bar{X}i$ = maximum of with member function.

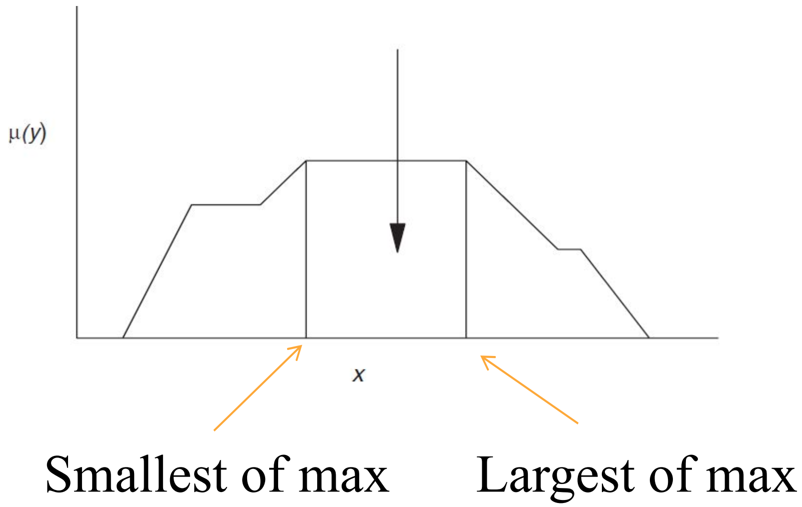
$\sum$ = algebraic sum.

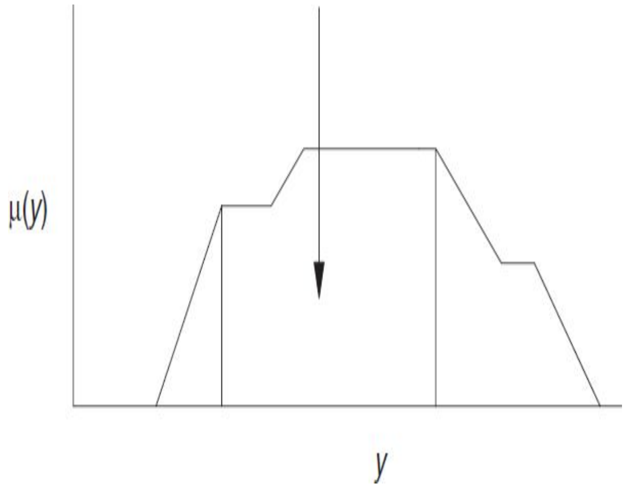
- **Mean max membership method:** This is known as middle of the maxima.

$$X^* = \frac{\sum_{i=1}^{n} \bar{xp}}{n}$$

Defuzzification Methods

- **Smallest of Maximum and Largest of Maximum**





- **Center of maxima:** In a multimode fuzzy region, the center-of-maxima technique finds the highest plateau and then the next highest plateau. The midpoint between the centers of these plateaus is selected

APPLICATIONS

- It is used in the **aerospace field** for altitude control of spacecraft and satellite.
- It has used in the **automotive system** for speed control, traffic control.
- It is used for **decision making support systems and personal evaluation** in the large company business.
- It has application in **chemical industry** for controlling the pH, drying, chemical distillation process.
- Fuzzy logic are used in **Natural language processing** and various intensive applications in **Artificial Intelligence**.
- Fuzzy logic are extensively used in **modern control systems such as expert systems**.

Thank you.