CS 480

Introduction to Artificial Intelligence

March 24, 2022

Announcements / Reminders

- **Programming Assignment #01:**
  - **due: ~~March 13th~~ ~~March 20th~~ TONIGHT, 11:00 PM CST**
- **Programming Assignment #02: POSTED**
- **New quiz: Monday**

- **Grading TA assignment:**

  https://docs.google.com/spreadsheets/d/1Cav_GBTGC7fLGzxuBCAUmEuJYPeF-HMLCYvwPbq8Fus/edit?usp=sharing

Plan for Today

- **Predicate / First-Order Logic - Resolution**
- **Quantifying and dealing with uncertainty**

Unification

Predicate logic inference rules require finding substitutions that make two different logical expressions look identical.

The process is called unification. A UNIFY **algorithm takes two sentences** p **and** q **and returns a unifier** θ **for them (a substitution) if one exists:**

$$\mathrm{UNIFY}(p, q) = \theta, \text{ where } \mathrm{SUBST}(\theta, p) = \mathrm{SUBST}(\theta, q)$$

Unification: Examples

UNIFY(sentenceA, sentenceB) = {unifier for sentenceA and sentenceB}

UNIFY(p, q) = {θ}

UNIFY(p, q) = {variable / unifying value}

Examples:

UNIFY(Knows(John, x), Knows(John, Jane)) = {x/Jane}

UNIFY(Knows(John, x), Knows(y, Bill)) = {x/Bill, y/John}

UNIFY(Knows(John, x), Knows(y, Mother(y))) = {x/Mother(John), y/John}

UNIFY(Knows(John, x), Knows(x, Elizabeth) = failure

({x/John, x/Elizabeth} is not possible)

Most General Unifier (MGU)

But.... ther can be multiple unifiers for a pair of sentences. Which one to choose?

Every UNIFIABLE pair of sentences has a SINGLE most general unifier that is unique.

UNIFY algorithm will find MGU.

Unification

function UNIFY(x, y, θ=*empty*) **returns** a substitution to make x and y identical, or *failure*
  if θ = *failure* **then return** *failure*
  else if $x = y$ **then return** θ
  else if VARIABLE?(x) **then return** UNIFY-VAR(x, y, θ)
  else if VARIABLE?(y) **then return** UNIFY-VAR(y, x, θ)
  else if COMPOUND?(x) **and** COMPOUND?(y) **then**
    return UNIFY(ARGS(x), ARGS(y), UNIFY(OP(x), OP(y), θ))
  else if LIST?(x) **and** LIST?(y) **then**
    return UNIFY(REST(x), REST(y), UNIFY(FIRST(x), FIRST(y), θ))
  else return *failure*

function UNIFY-VAR(var, x, θ) **returns** a substitution
  if $\{var/val\} \in \theta$ for some val **then return** UNIFY(val, x, θ)
  else if $\{x/val\} \in \theta$ for some val **then return** UNIFY(var, val, θ)
  else if OCCUR-CHECK?(var, x) **then return** *failure*
  else return add $\{var/x\}$ to θ

Predicate Logic Resolution: Example

Consider following sentences in English

A. **Everyone who loves all animals is loved by someone**
B. **Anyone who kills an animal is loved by no one**
C. **Jack Loves all animals**
D. **Either Jack or Curiosity killed the cat, who is named Tuna**

Q. **Did Curiosity kill the cat?**

FOL: The Resolution Inference Rule

Two clauses, which are assumed to be standardized apart, so that they share no variables, can be resolved if they contain complementary literals:

- **Propositional literals are complementary if one is the negation of the other**
- **Predicate logic literals are complimentary if one unifies with the negation of the other**

$$\frac{(l_1 \vee \ldots \vee l_k),\ (m_1 \vee \ldots \vee m_n)}{\mathrm{SUBST}(\theta, l_1 \vee \ldots \vee l_{i-1} \vee l_{i+1} \vee \ldots \vee l_k \vee m_1 \vee \ldots \vee m_{j-1} \vee m_{j+1} \vee \ldots \vee m_n)}$$

where $\theta = \mathrm{UNIFY}(l_{i-1}, m_j)$**.**

FOL: The Resolution Inference Rule

For example, the following two clauses:

$$[\text{Animal}(F(x)) \lor \text{Loves}(G(x), x)]$$ **and**

$$[\neg\text{Loves}(u, v) \lor \neg\text{Kills}(u, v)]$$

can be resolved by eliminating complementary literals

$$\text{Loves}(G(x), x) \text{ and } \neg\text{Loves}(u, v)$$

with the unifier

$$\theta = \{u/G(x), v/x\},$$

to produce the resolvent clause:

$$[\text{Animal}(F(x)) \lor \neg\text{Kills}(G(x), x)]$$

Predicate Logic Resolution: Example

Now, let's turn them into predicate logic sentences/KB:

A. $\forall x\ [\forall y\ (Animal(y) \Rightarrow Loves(x, y))] \Rightarrow [\exists y\ Loves(y, x)]$

B. $\forall x\ [\exists z\ (Animal(z) \land Kills(x, z))] \Rightarrow [\forall y\ \neg Loves(y, x)]$

C. $\forall x\ [Animal(x) \Rightarrow Loves(Jack, x)]$

D. $Kills(Jack, Tuna) \lor Kills(Curiosity, Tuna)$

E. $Cat(Tuna)$

F. $\forall x\ [Cat(x) \Rightarrow Animal(x)]$

Q. $Kills(Curiosity, Tuna)$**, so** $\neg Q \equiv \neg Kills(Curiosity, Tuna)$

Predicate Logic Resolution: Example

Let's turn them into predicate logic CNF sentences/KB:

A1. $(Animal(F(x)) \lor Loves(G(x), x))$ **(A1 and A2 related)**

A2. $(\neg Loves(x, F(x)) \lor Loves(G(x), x))$

B. $(\neg Loves(y, x) \lor \neg Animal(z) \lor \neg Kills(x, z))$

C. $(\neg Animal(x) \lor Loves(Jack, x))$

D. $(Kills(Jack, Tuna) \lor Kills(Curiosity, Tuna))$

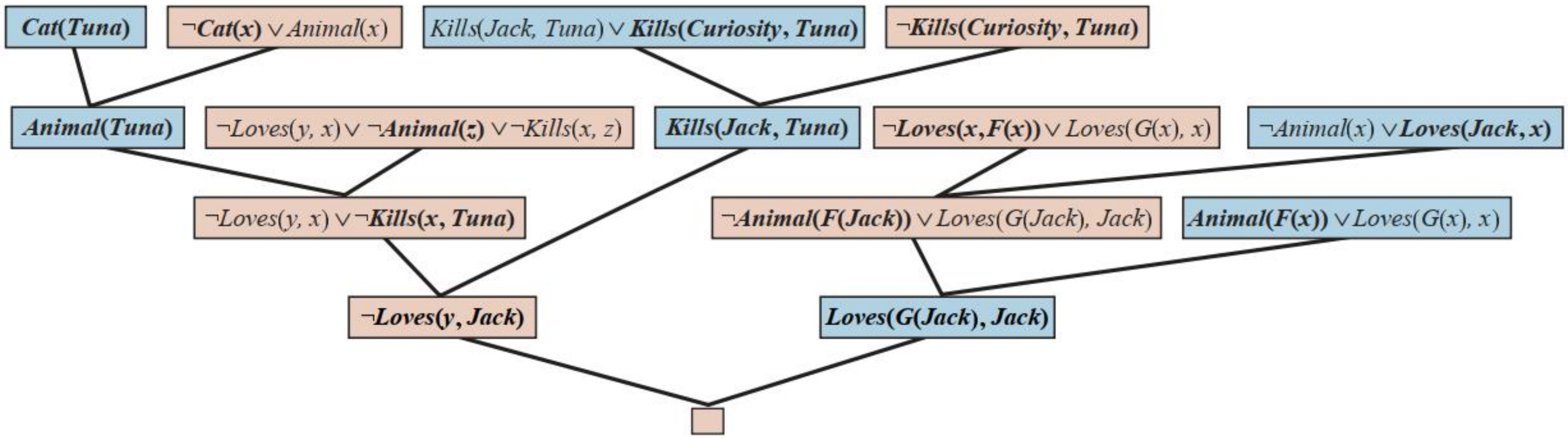
E. $(Cat(Tuna))$

F. $(\neg Cat(x) \lor Animal(x))$

Q. $Kills(Curiosity, Tuna)$, **so** $\neg Q \equiv (\neg Kills(Curiosity, Tuna))$

Predicate Logic Resolution: Example

Resolution process with substitutions:



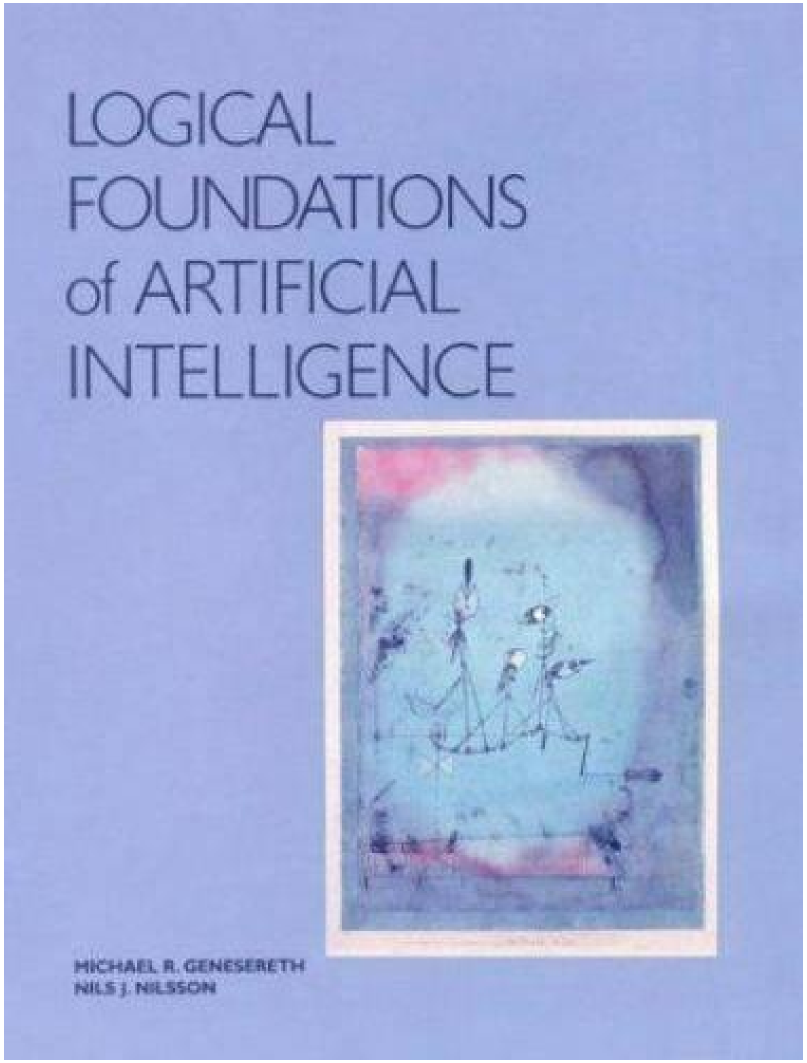
Notice the use of factoring in derivation of the clause(Loves(G(Jack), Jack)**)**

If you want more on Logic...

Michael Genesereth, Nils J. Nilsson

“Logical foundations of artificial intelligence”

Elsevier 1978



Probability Theory: Need to Know

- What is an **event** A?
- What is the **probability of event** A occuring ($P(A)$)?
- What is a **random variable** X?
- What is the **probability distribution** for X?
- What is the **probability density function** for X?
- What are the **expectation** and **variance** of X?

- Check out https://seeing-theory.brown.edu/ for a refresher

Probability Theory: Need to Know

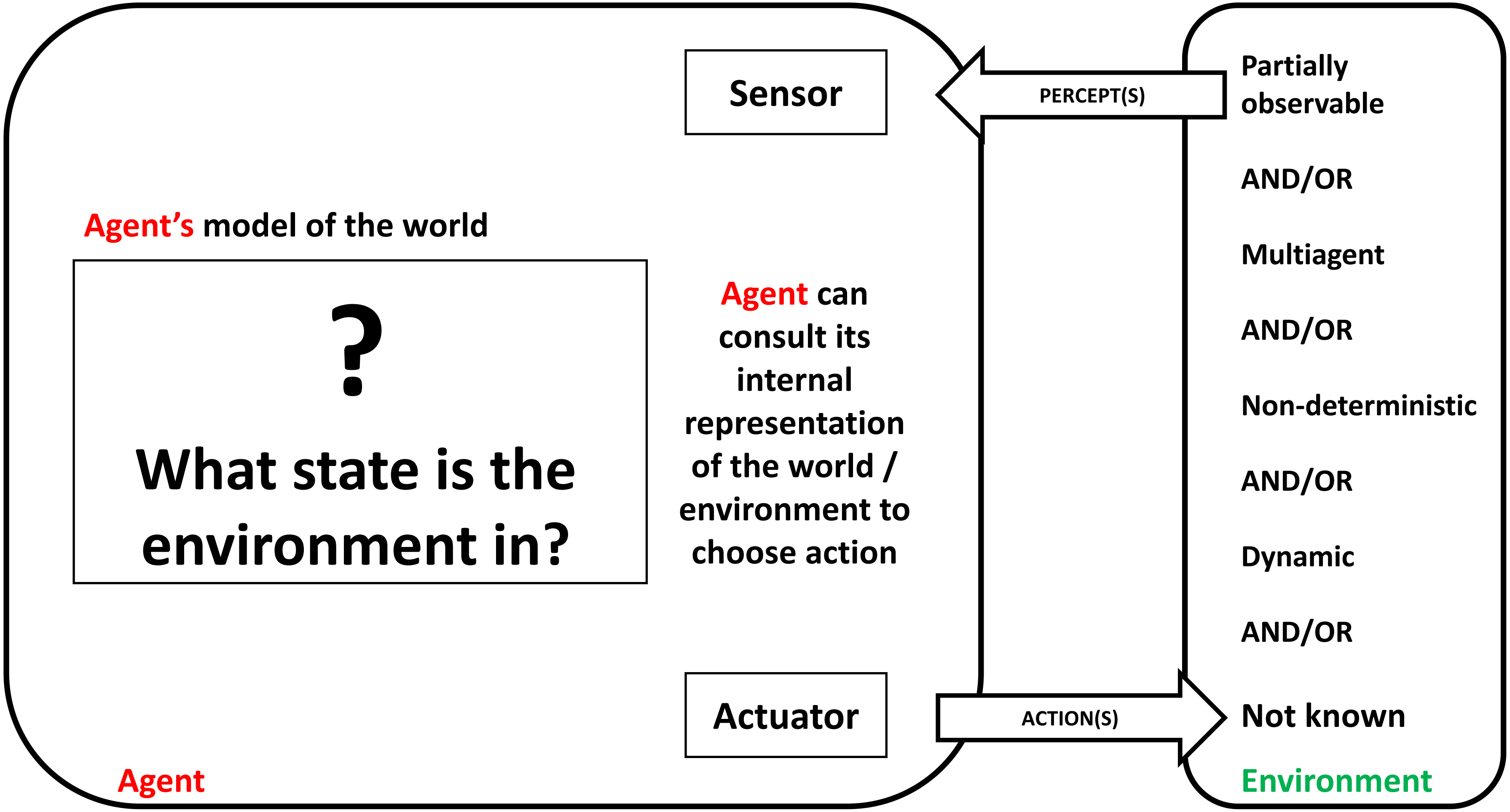
- P(sure event) = 1 and
- P(impossible event) = 0
- **If** A, B **are exclusive events:** $P(A \vee B) = P(A) + P(B)$
- **If** A, B **are complementary events:** $P(A) + P(\neg A) = 1$
- **If** A, B **are arbitrary events:**

$$P(A \vee B) = P(A) + P(B) - P(A \wedge B)$$

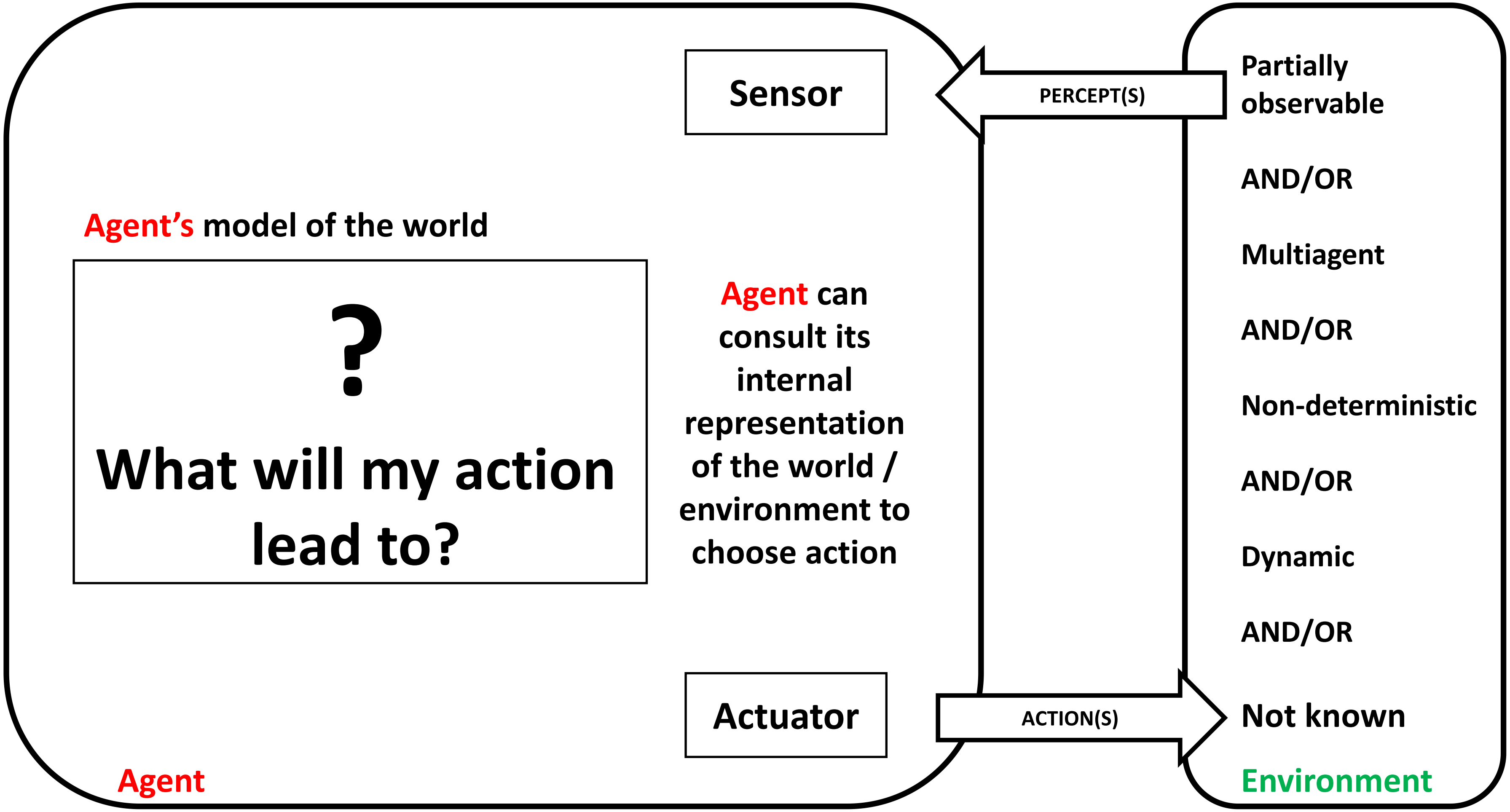
- **If** $A \subseteq B$**, it is true that** $P(A) \leq P(B)$
- **If** $A_1, A_2, \ldots, A_n$ **are elementary events, then:**

$$\sum_{i=1}^{n} P(A_i) = 1$$

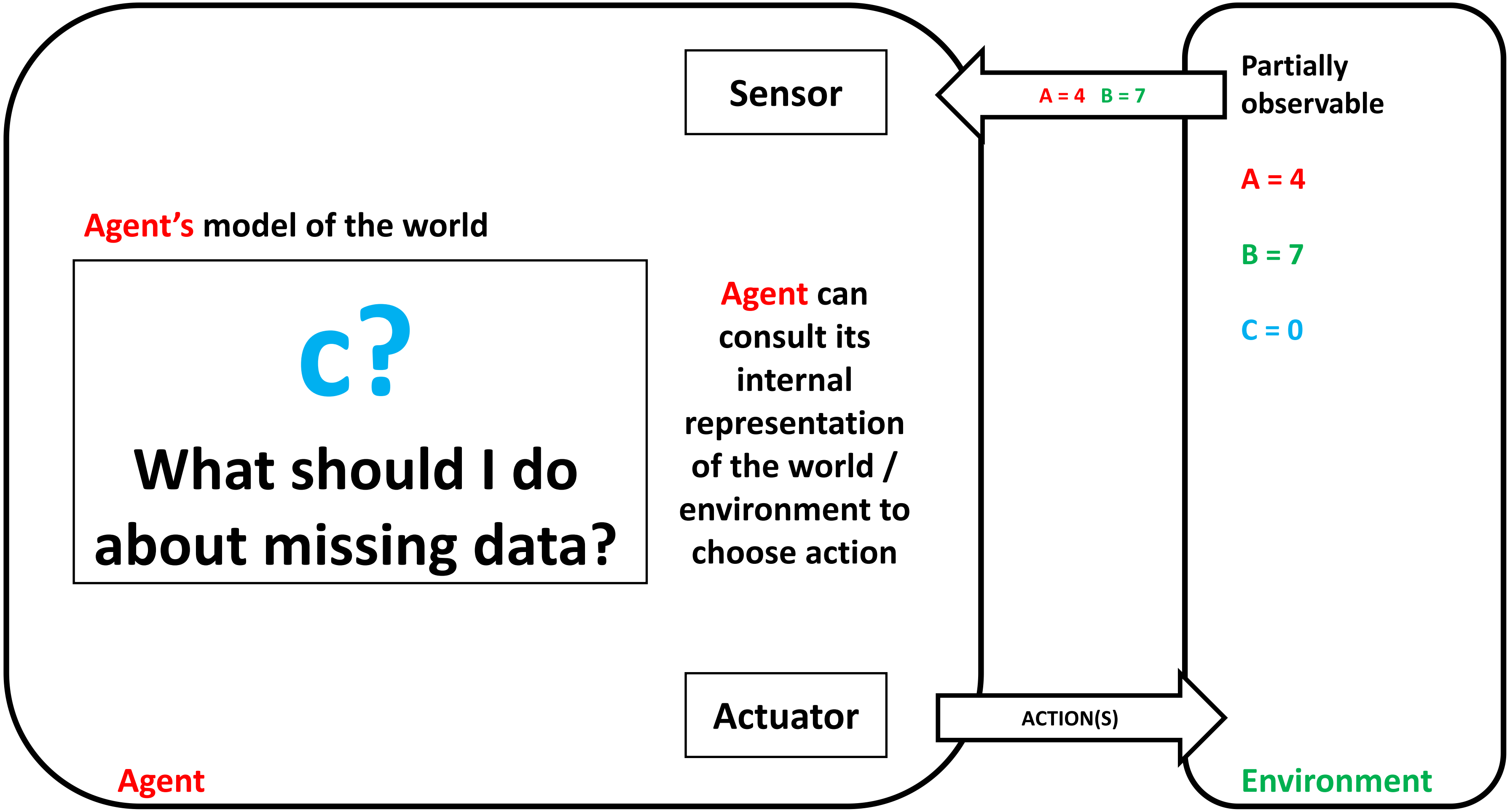
Agents and Uncertainty



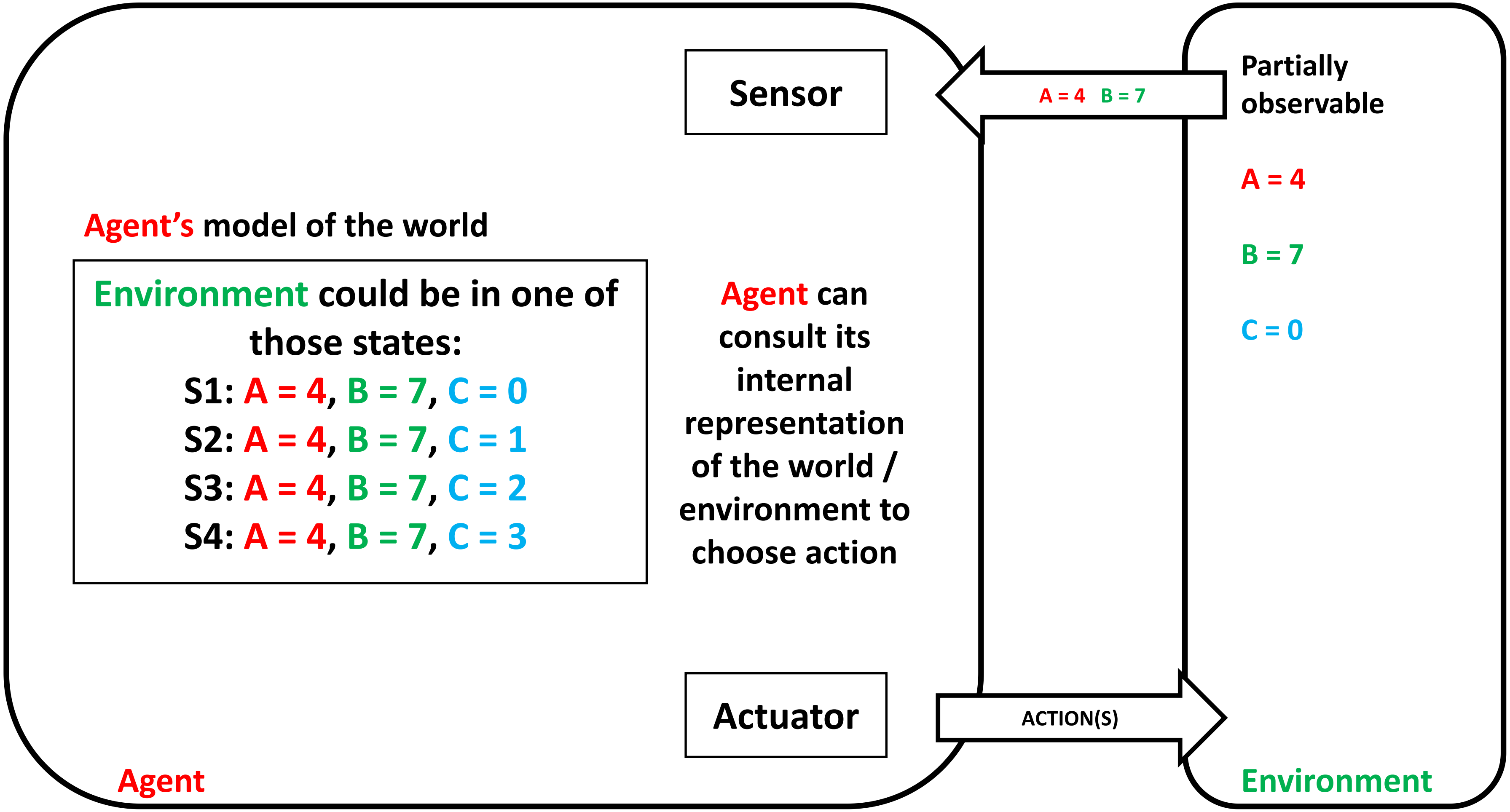
Agents and Uncertainty



Agents and Uncertainty



Agents and Belief State



Assume: $D_C = \{0,1,2,3\}$

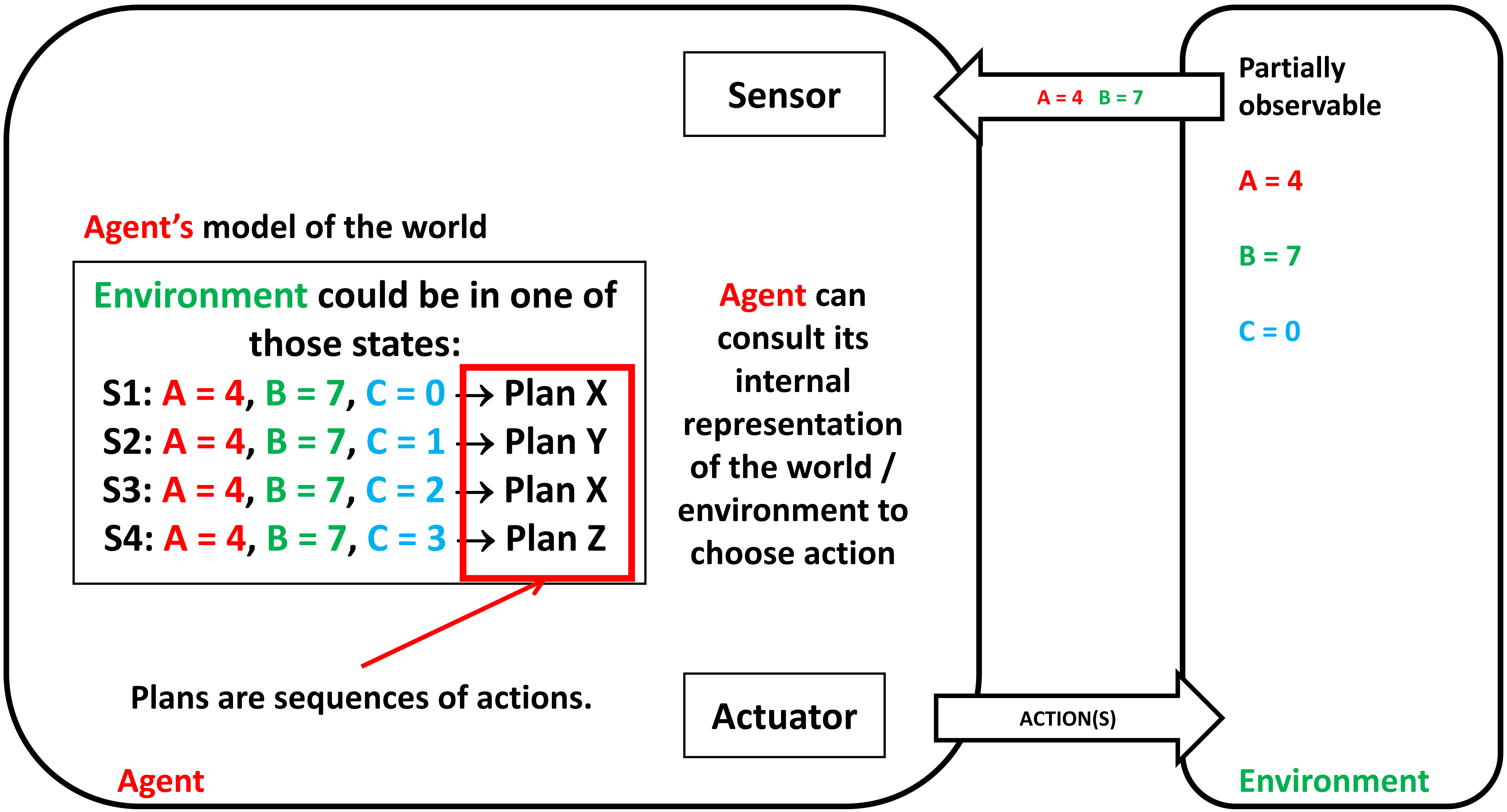
Agent Belief State

Belief state: a **set of all possible environment states that the agent can be in** and needs to keep track of to handle uncertainty.

Problems:

- agent needs to consider every possible state some are going to be unlikely
- agent needs plans for every eventuality
- there may be no known plan, agent needs to act

Agents and Belief State



Assume: $D_C = \{0,1,2,3\}$

Decision Theory

- **Decisions: every plan (actions) leads to an outcome (state)**
- **Agents have preferences (preferred outcomes)**
- **Preferences $\rightarrow$ outcome utilities**
- **Agents have degrees of belief (probabilities) for actions**

Decision theory = probability theory + utility theory

Frequentist versus Causal Perspective

- **Frequentist view:**

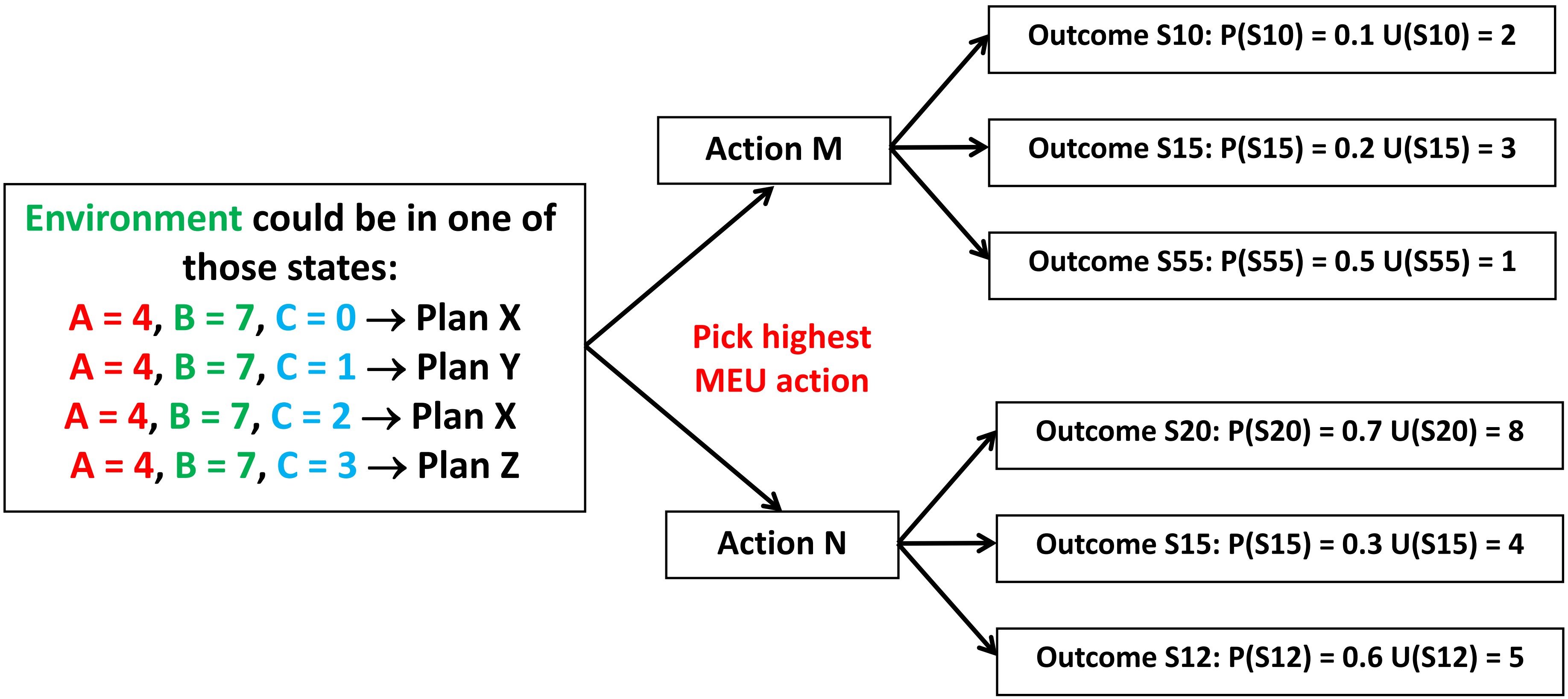
  Probability represents long-run frequencies of repeatable events.

- **Causal perspective:**

  Probability is a measure of belief.

Maximum Expected (Average) Utility

$$MEU(M) = \frac{P(S10) * U(S10) + P(S15) * U(S15) + P(S55) * U(S55)}{3}$$



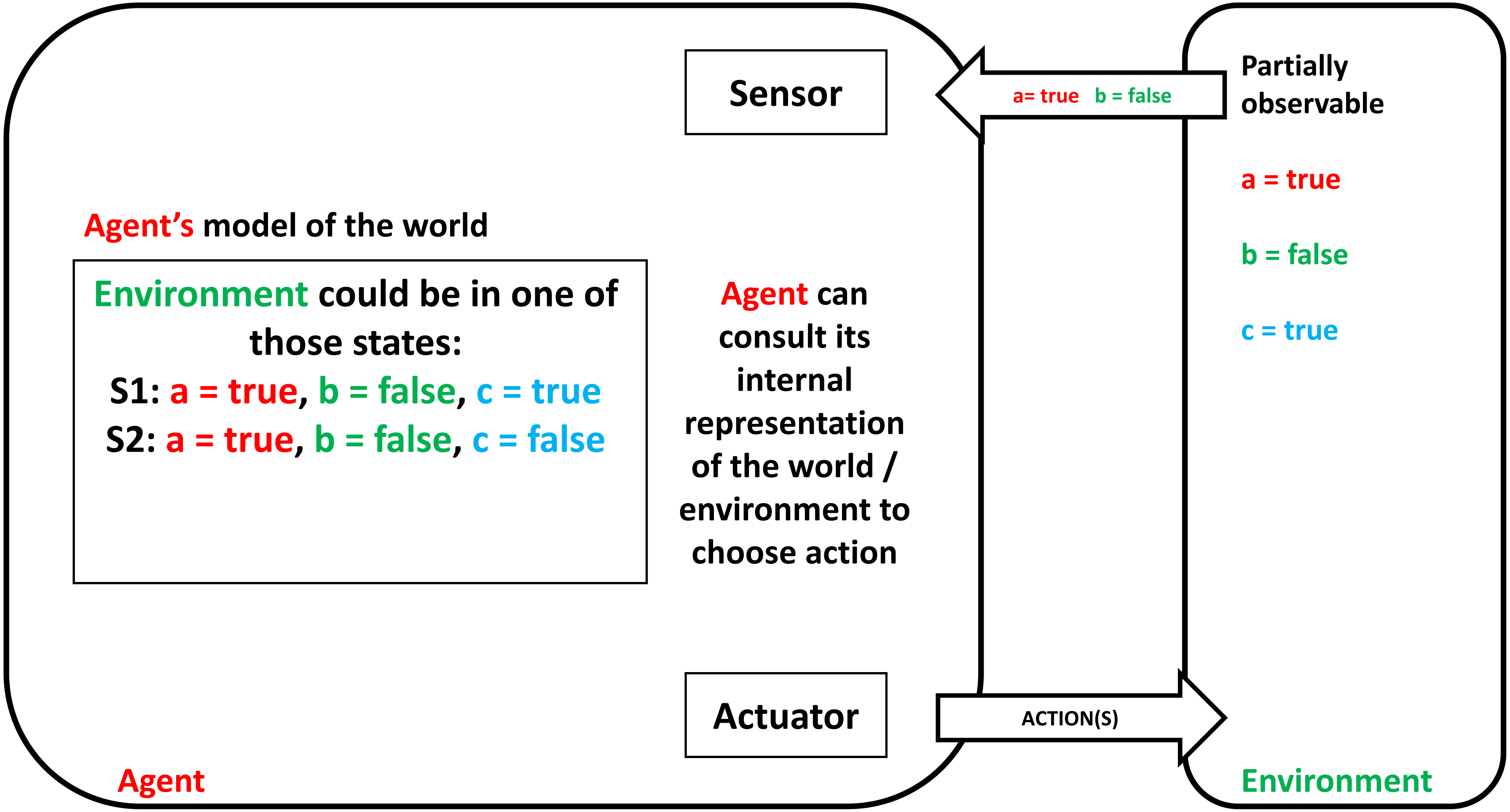
$$MEU(N) = \frac{P(S20) * U(S20) + P(S15) * U(S15) + P(S12) * U(S12)}{3}$$

Decision-theoretic Agent

function DT-AGENT(*percept*) **returns** an *action*
  persistent: *belief_state*, probabilistic beliefs about the current state of the world
      action, the agent's action

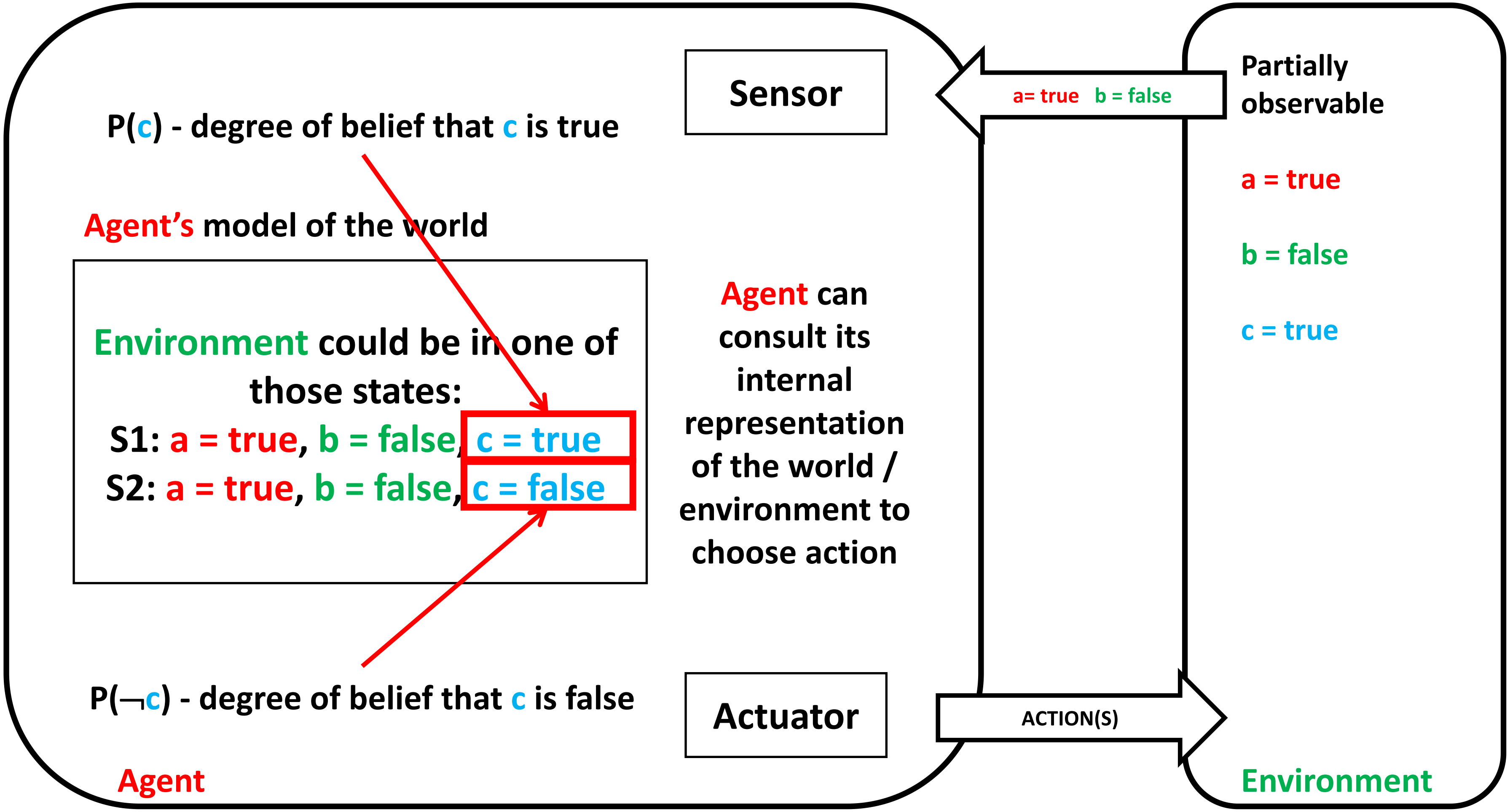
  update *belief_state* based on *action* and *percept*
  calculate outcome probabilities for actions,
    given action descriptions and current *belief_state*
  select *action* with highest expected utility
    given probabilities of outcomes and utility information
  return *action*

Agents and Belief State



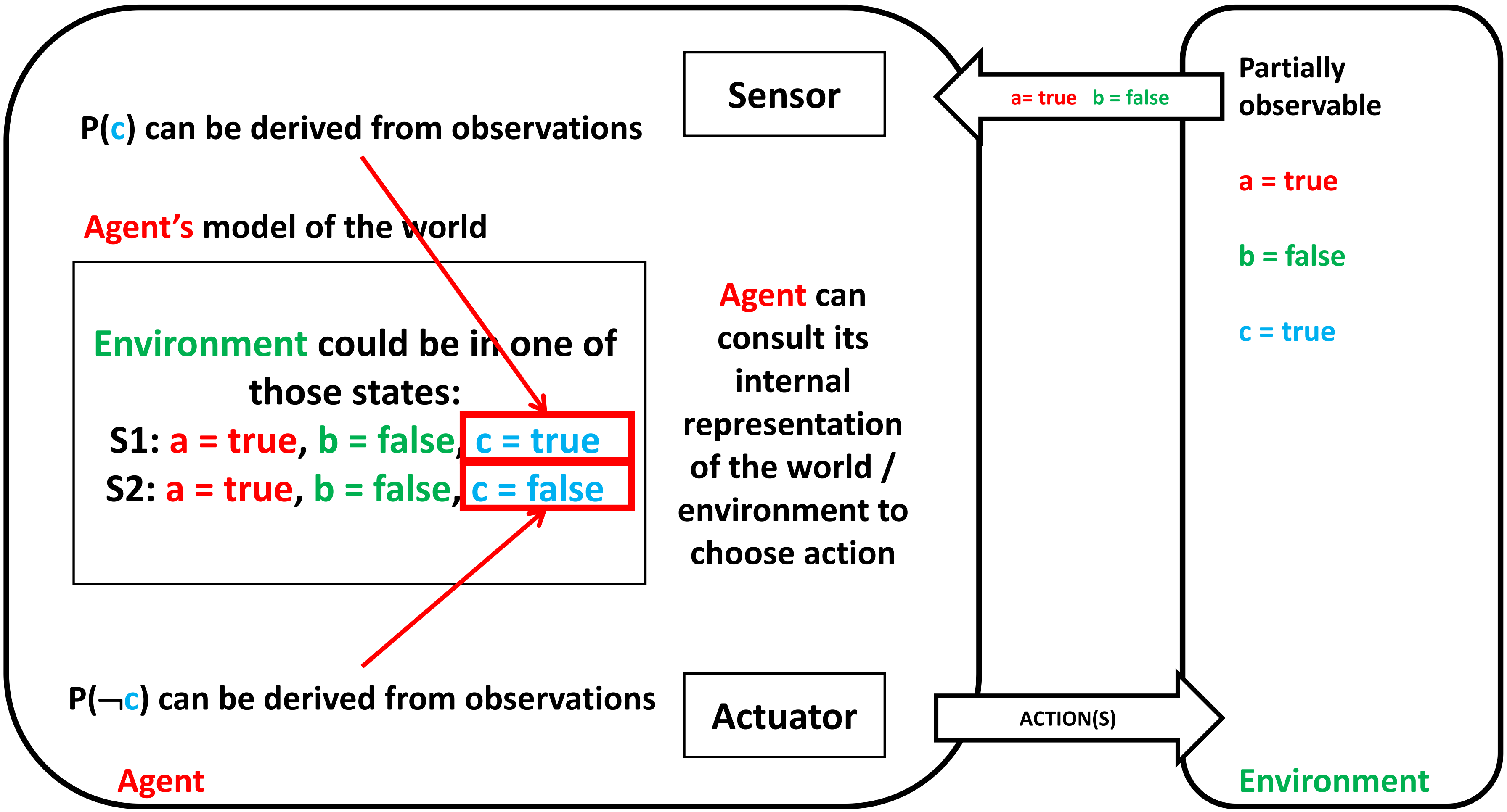
Assume: D_c = {true, false}

Agents and Belief State



Assume: D_c = {true, false}

Agents and Belief State



Assume: D_c = {true, false}

Relationships in Probability Language

- **Likelihood:**

"Tim is *more likely* to fly than to walk."

- **Conditioning:**

"*If* Tim is sick, he can't fly."

- **Relevance:**

"Whether Tim flies *depends on whether* he is sick."

- **Causation:**

"Being sick *caused* Tim's inability to fly."

Probability Theory and Propositions

Assume that A **and** B **are sentences in propositional logic.**

- $P(\top) = 1$
- $P(\bot) = 0$
- $P(A \vee B) = P(A) + P(B)$ **if** $\neg(A \wedge B)$ **is a tautology**
- $P(A) + P(\neg A) = 1$
- $P(A) = P(B)$ **if** $(A \Leftrightarrow B)$ **is a tautology (logical equivalence)**
- $0 \leq P(A)$ **for any sentence** A

Probability Model

A fully specified probability model associates a numerical probability $\mathrm{P}(\omega)$ with each possible world (assume there is a finite number of such worlds):

$$0 \leq P(\omega) \leq 1 \; for\; every \sum_{\omega \in \Omega} P(\omega) = 1$$

Propositions and Probabilities

The probability associated with a proposition A is defined to be the sum of the probabilities of all the worlds in which it holds:

$$For\ any\ proposition\ A, P(A)\ \sum_{\omega \in A} P(\omega)$$

Prior (Unconditional) Probabilities

Degree of belief that some proposition A is true *in the absence of any other related information* is called unconditional or prior probability (or “prior” for short) $P(A)$.

Examples:

$$P(\text{isRaining})$$

$$P(\text{dieRoll} = 5)$$

$$P(\text{CS480FinalGrade} = \text{‘A’})$$

$$P(\text{toothache})$$

Conditioning

Conditioning is a process of revising beliefs based on new evidence e:

- **start by taking all background information (prior probabilities) into account**
- **if new evidence e is acquired, a conditional probability of some proposition A given evidence e can be calculated (posterior probability):** $P(A \mid e)$

Posterior (Conditional) Probabilities

Typically, there is going to be some information, called evidence e, that affects our degree of belief about some proposition A being true. This allows us to also consider conditional or posterior probability (or "posterior" for short) $P(A \mid e)$.

Examples ($P(A \text{ given } e)$):

$$P(\text{isRaining} \mid \text{cloudy})$$

$$P(\text{CS480FinalGrade} = \text{'A'} \mid \text{CS480PA1Score} > 80)$$

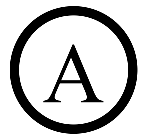
$$P(\text{cavity} \mid \text{toothache})$$

Evidence e

Evidence e **rules out possible worlds incompatible with** e**.**

Prior vs. Posterior Probabilities

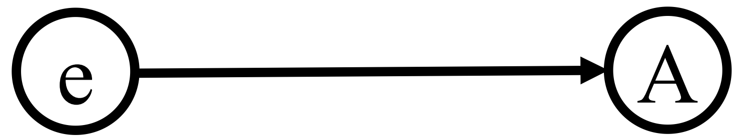
Prior Probability



$$P(A)$$

BTW: it is also $P(A \mid T)$

Posterior Probability



$$P(A \mid e)$$

Conditional Probability

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

where $\mathrm{P(B)} > 0$

Conditional Probability

$$\boldsymbol{P(A \mid evidence) = \frac{P(A \wedge evidence)}{P(evidence)}}$$

where P(evidence) > 0

Conditional Probability (Product Rule)

$$P(A \wedge B) = P(A \mid B) * P(B)$$

Conditional Probability (Product Rule)

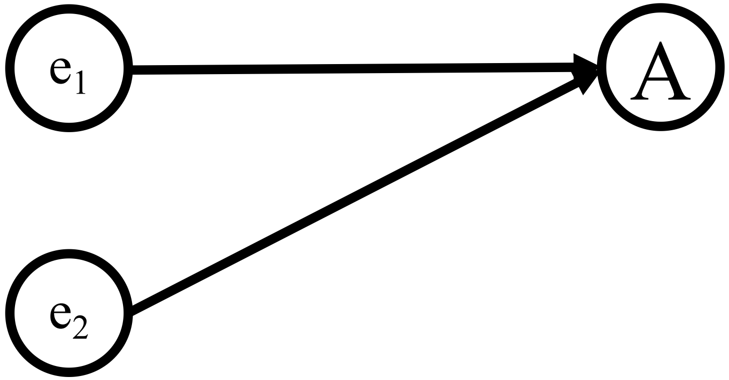
$$P(A \wedge evidence) = P(A \mid evidence) * P(evidence)$$

Prior vs. Posterior Probabilities

Prior Probability

$$P(A)$$

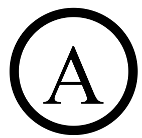
Posterior Probability



$$P(A \mid e_1 \wedge e_2)$$

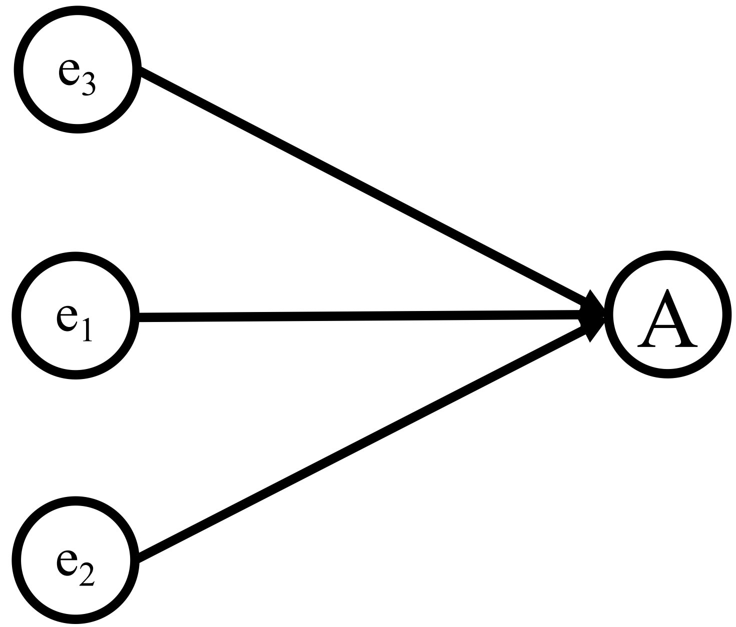
Prior vs. Posterior Probabilities

Prior Probability



$$P(A)$$

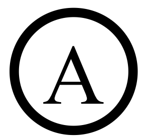
Posterior Probability



$$P(A \mid e_1 \wedge e_2 \wedge e_3)$$

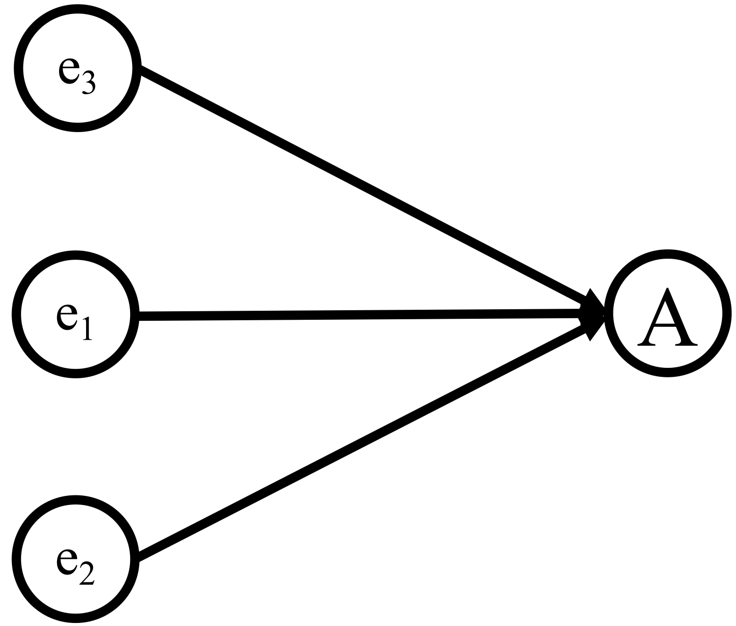
Prior vs. Posterior Probabilities

Prior Probability



$$P(A)$$

Posterior Probability



$$P(A \mid \text{parents}(A))$$

Marginal Probability

Marginal probability: the probability of an event occurring $P(A)$.

It may be thought of as an unconditional probability.

It is not conditioned on another event.

Joint Probability

The probability of event A and event B occurring (or more than two events). It is the probability of the intersection of two or more events.

$$\boldsymbol{P(A \wedge B)}$$

For any propositions $\boldsymbol{f_1, f_2, \ldots, f_n}$:

$$\boldsymbol{P(f_1 \wedge f_2 \wedge \ldots \wedge f_n)}$$

Chain Rule

Conditional probabilities can be used to decompose conjunctions using the chain rule. For any propositions $f_1, f_2, \ldots, f_n$:

$$P(f_1 \land f_2 \land \ldots \land f_n) =$$
$$P(f_1) *$$
$$P(f_2 \mid f_1) *$$
$$P(f_3 \mid f_1 \land f_2) *$$
$$\ldots$$
$$P(f_n \mid f_1 \land \ldots \land f_{n-1}) =$$
$$= \prod_{i=1}^{n} P(f_i \mid f_1 \land \ldots \land f_{i-1})$$

Chain Rule

Conditional probabilities can be used to decompose conjunctions using the chain rule. For any propositions $f_1, f_2, \ldots, f_n$ **:**

$$P(f_1 = x_1 \wedge f_2 = x_2 \wedge \ldots \wedge f_n = x_n) =$$
$$P(f_1 = x_1) *$$
$$P(f_2 \mid f_1 = x_1) *$$
$$P(f_3 \mid f_1 = x_1 \wedge f_2 = x_2) *$$
$$\ldots$$
$$P(f_n = x_n \mid f_1 = x_1 \wedge \ldots \wedge f_{n-1} = x_{n-1}) =$$
$$= \prod_{i=1}^{n} P(f_i = x_i \mid f_1 = x_1 \wedge \ldots \wedge f_{i-1} = x_{i-1})$$

Bayes' Rule

$$P(A \mid B) = \frac{P(B \mid A) * P(A)}{P(B)}$$

Bayes' Rule

$$P(cause \mid effect) = \frac{P(effect \mid cause) * P(cause)}{P(effect)}$$

Bayes' Rule

$P(cause \mid effect)$ **diagnostic direction relation**

$$P(cause \mid effect) = \frac{P(effect \mid cause) * P(cause)}{P(effect)}$$

$P(effect \mid cause)$ **causal direction relation**

Bayes' Rule

$P(disease \mid symptoms)$ **diagnostic direction relation**

$$P(disease \mid symptoms) = \frac{P(symptoms \mid disease) * P(disease)}{P(symptoms)}$$

$P(symptoms \mid disease)$ **causal direction relation**

Bayes' Rule

$$P(cause \mid effect) = \frac{P(effect \mid cause) * P(cause)}{P(effect)}$$

Problem: a single card is drawn from a standard deck of cards. What is the probability that we drew a queen if we know that a face card (J, Q, K) was drawn?

$$P(queen \mid face) = \frac{P(face \mid queen) * P(queen)}{P(face)}$$

$$P(queen \mid face) = \frac{1 * 4 / 52}{12 / 52} = \frac{1}{3}$$

Bayes' Rule

$$P(cause \mid effect) = \frac{P(effect \mid cause) * P(cause)}{P(effect)}$$

Problem: Calculate probability that a patient has meningitis if a patient has stiff neck. Meningitis is a cause of neck stiffness in 70% of cases, probability of having meningitis is 1/50000. Stiff neck happens to 1% of patients.

$$P(m \mid s) = \frac{P(s \mid m) * P(m)}{P(s)}$$

$$P(m \mid s) = \frac{0.7 * 1/50000}{0.01} = 0.0014$$

Independence

Assume that the knowledge of the truth of one proposition Y, does not affect the agent's belief in another proposition, X, in the context of other propositions Z. We say that X is independent of Y given Z.

Conditional Independence

Random variable X is conditionally independent of random variable Y given Z if for all $x \in Dx$, for all $y \in Dy$, and for all $z \in Dz$, such that

$$P(Y = y \wedge Z = z) > 0 \textbf{ and } P(Y = y' \wedge Z = z) > 0$$

$$P(X = x \mid Y = y \wedge Z = z) = P(X = x \mid Y = y' \wedge Z = z)$$

In other words, given a value of Z, knowing Y's value DOES NOT affect your belief in the value of X.

Conditional Independence

The following four statements are equivalent as long as conditional probabilities:

1. X **is conditionally independent of** Y **given** Z
2. Y **is conditionally independent of** X **given** Z
3. $P(X \mid Y, Z) = P(X \mid Z)$
4. $P(X, Y \mid Z) = P(X \mid Z) * P(Y \mid Z)$

Bayesian (Belief) Network

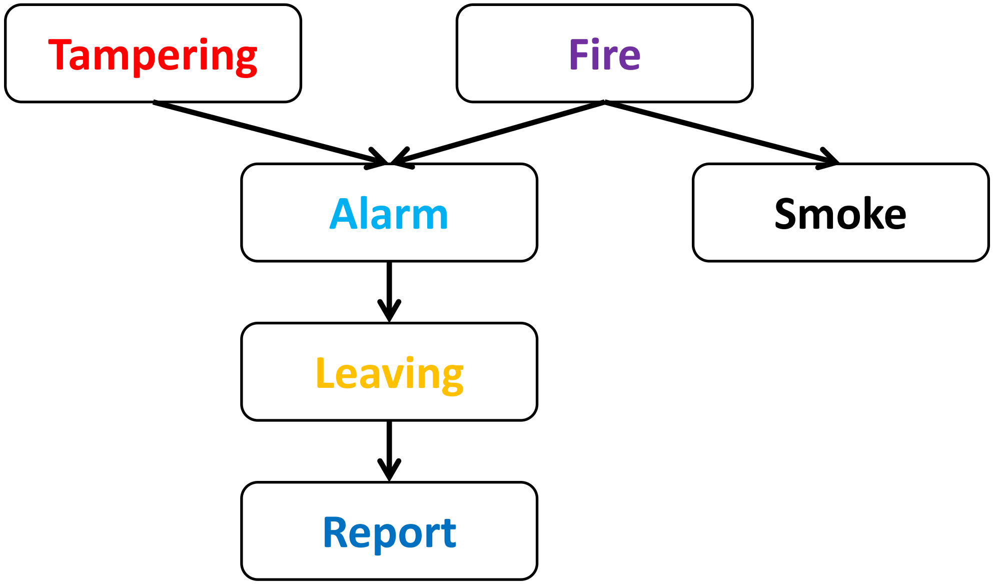
A Bayesian belief network describes the joint probability distribution for a set of variables.

A Bayesian network is an acyclic, directed graph (DAG), where the nodes are random variables (propositions). There is an edge (arc) from each elements of $\mathrm{parents}(X_i)$ into X_i. Associated with the Bayesian network is a set of conditional probability distributions - the conditional probability of each variable given its parents (which includes the prior probabilities of those variables with no parents).

Consists of:

- **a graph (DAG) with nodes corresponding to random variables**
- **a domain for each random variable**
- **a set of conditional probability distributions $P(X_i \mid \mathrm{parents}(X_i))$**

Bayesian (Belief) Network: Example



Random Variables (Propositions):

- **Tampering: true if the alarm is tampered with**
- **Fire: true if there is a fire**
- **Alarm: true if the alarm sounds**
- **Smoke: true if there is smoke**
- **Leaving: true if people leaving the building at once**
- **Report: true if someone who left the building reports fire**

Domain for all variables: $\{\text{true, false}\}$

NOTE: RVs don't have to be Boolean