CS 5/7320 Artificial Intelligence

Making Simple Decisions AIMA Chapter 16

Introduction slides by Michael Hahsler

Decision network slides by Dan Klein and Pieter Abbeel

Online Material

Decision-theoretic Agents (=Utility-based Agent)

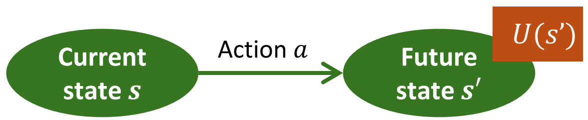
Recap

- **Agents based on logic**: Cannot deal with uncertainty, conflicting goals, etc.
- **Goal-based agents**: Can only assign goal/not goal to states and find goal states.

- **Utility-based agents**
  - Assign a utility value to each state.
  - Utility is related to the external performance measure (see PEAS).
  - A rational agent optimizes the expected utility (i.e., is utility-based).
  - Decisions are made using decision theory.

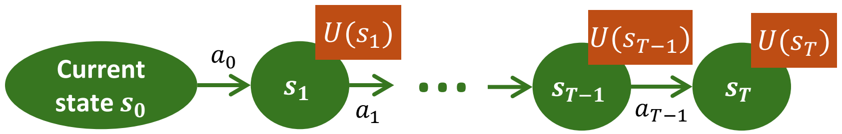
Decision theory =
Probability theory (evidence & belief)
+
Utility theory (want)

Simple vs. Complex Decisions



- We make the same decision frequently + making it once does not affect future decisions. This means we have an **episodic environment**.
- May have a **stochastic** environment (e.g., with non-deterministic actions or probabilistic transitions).
- May be **partially observable**.

We focus on making simple decisions for now!



- **Sequential decision making**: The agent's utility depends on a sequence of decisions.
- Search, planning and playing games we have covered so far are such problems.
- To solve this with decision theory requires different methods: **Markov Decision Processes (MDP)**

Utility

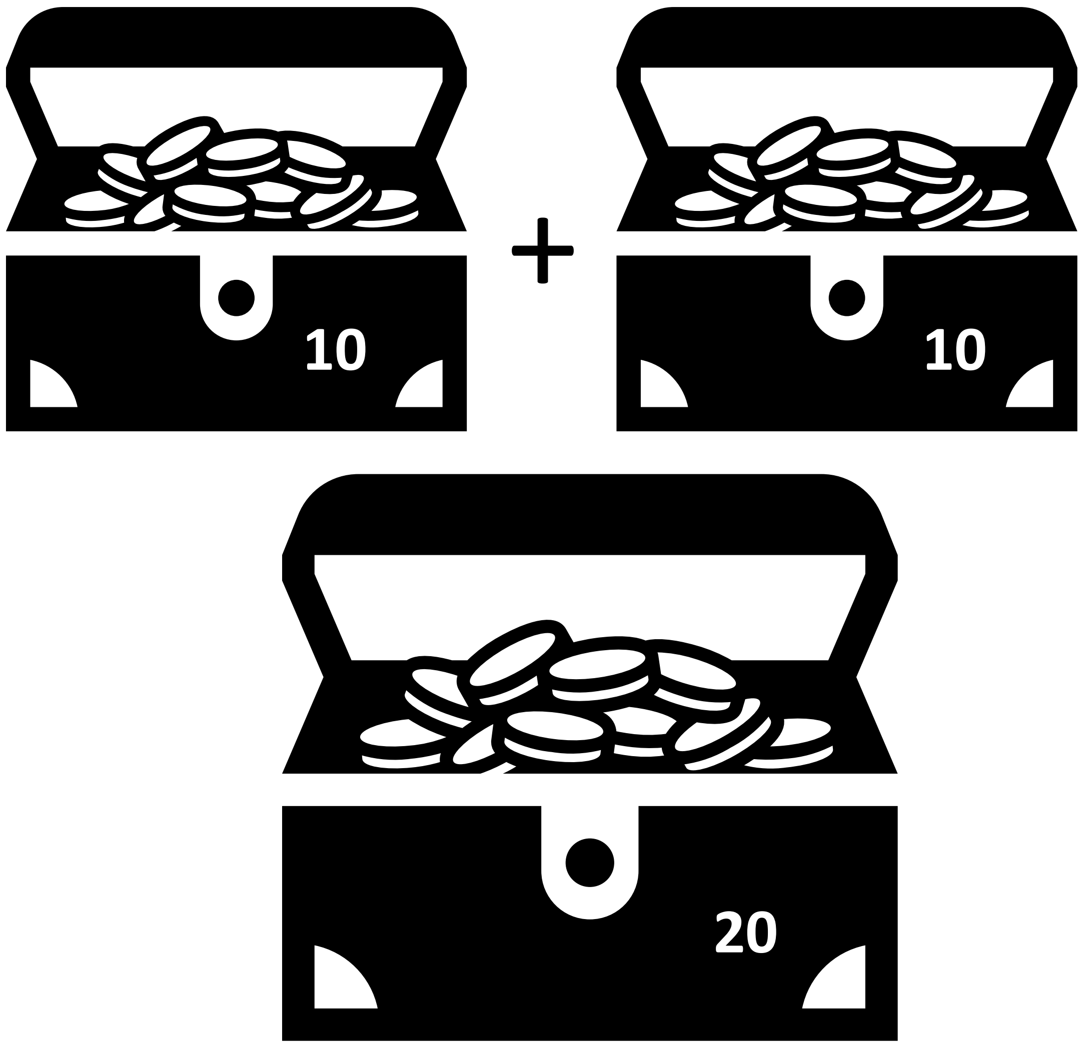
- A utility function $U(s)$ expresses the desirability of being in state s.
- Utility functions are derived from preferences:

$$U(A) > U(B) \Leftrightarrow A \succ B$$

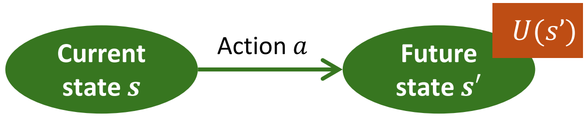
and

$$U(A) = U(B) \Leftrightarrow A \sim B$$

- It is often enough to know a **ordinal utility function** representing a **ranking** of states to make decisions like move to the better state.
- To use expectation, we need a **cardinal utility function** where the number represents levels of absolute satisfaction. That is $2 \times U$ is twice as good as U!



Expected Utility of an Action Under Uncertainty



We need:

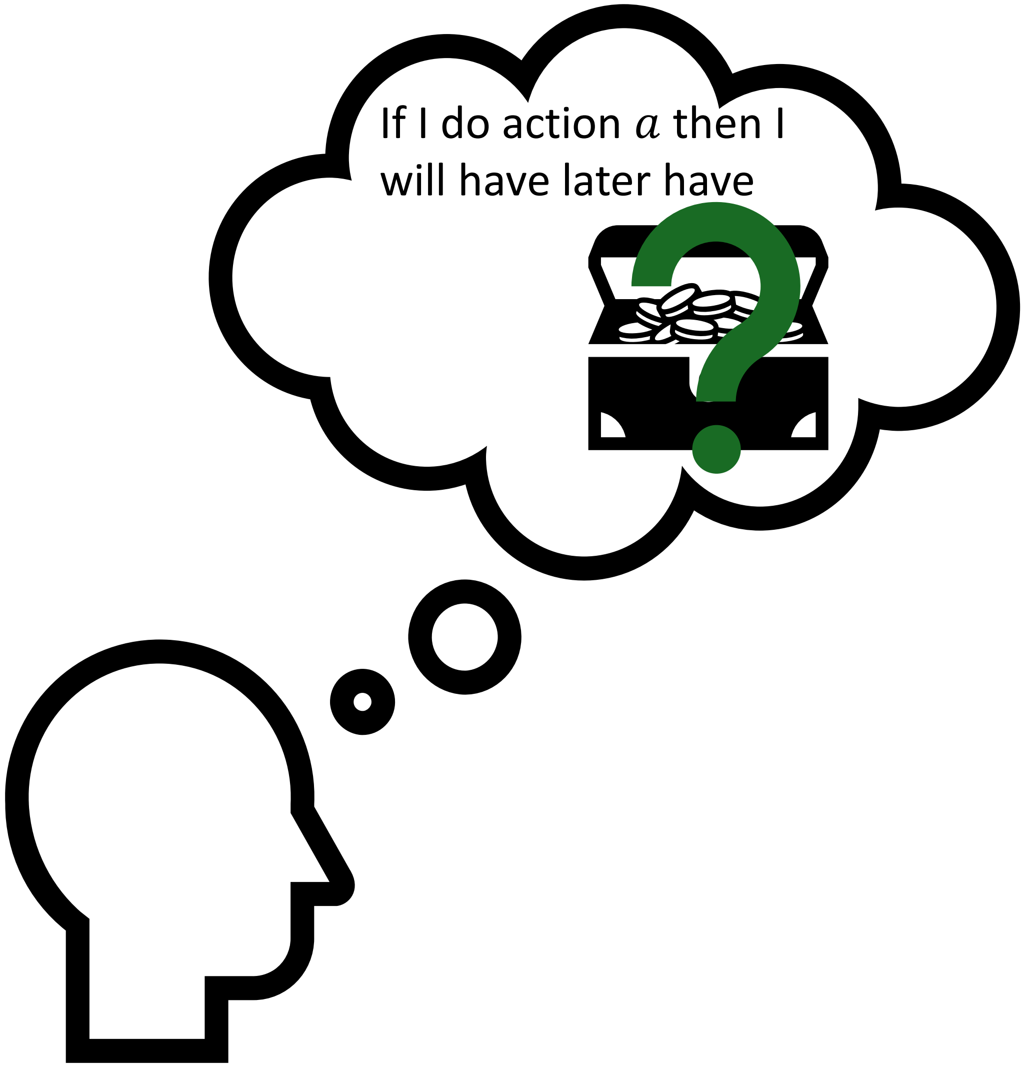
- The probability distribution $P(s)$, that the current state is s.
- The transition model specified as transition probabilities $P(s'|s,a)$ of getting from s to s' given action a.
- A **cardinal utility** function $U(s)$.

The probability that action a will get us from s to a future state s' is

$$P(s') = \sum_{s} P(s)\, P(s'|s,a)$$

The expected utility of action a over all possible future states is

$$EU(a) = \sum_{s'} P(s')\, U(s') = \sum_{s'} \sum_{s} P(s)\, P(s'|s,a)\, U(s')$$



Principle of Maximum Expected Utility (MEU)

Given the expected utility of an action

$$EU(a) = \sum_{s'} \sum_{s} P(s)\, P(s'|s,a)\, U(s')$$

Choose the action that maximizes the expected utility:

$$a^* = \text{argmax}_a\, EU(a)$$

Issue:

- $P(s'|s,a)$ will be a very large table if we have many states .

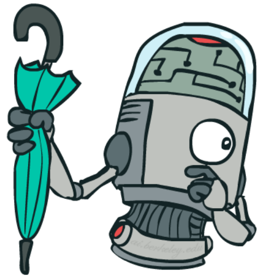
Possible solution:

- Bayes Nets with a factored state representation considering independence between variable describing the state.

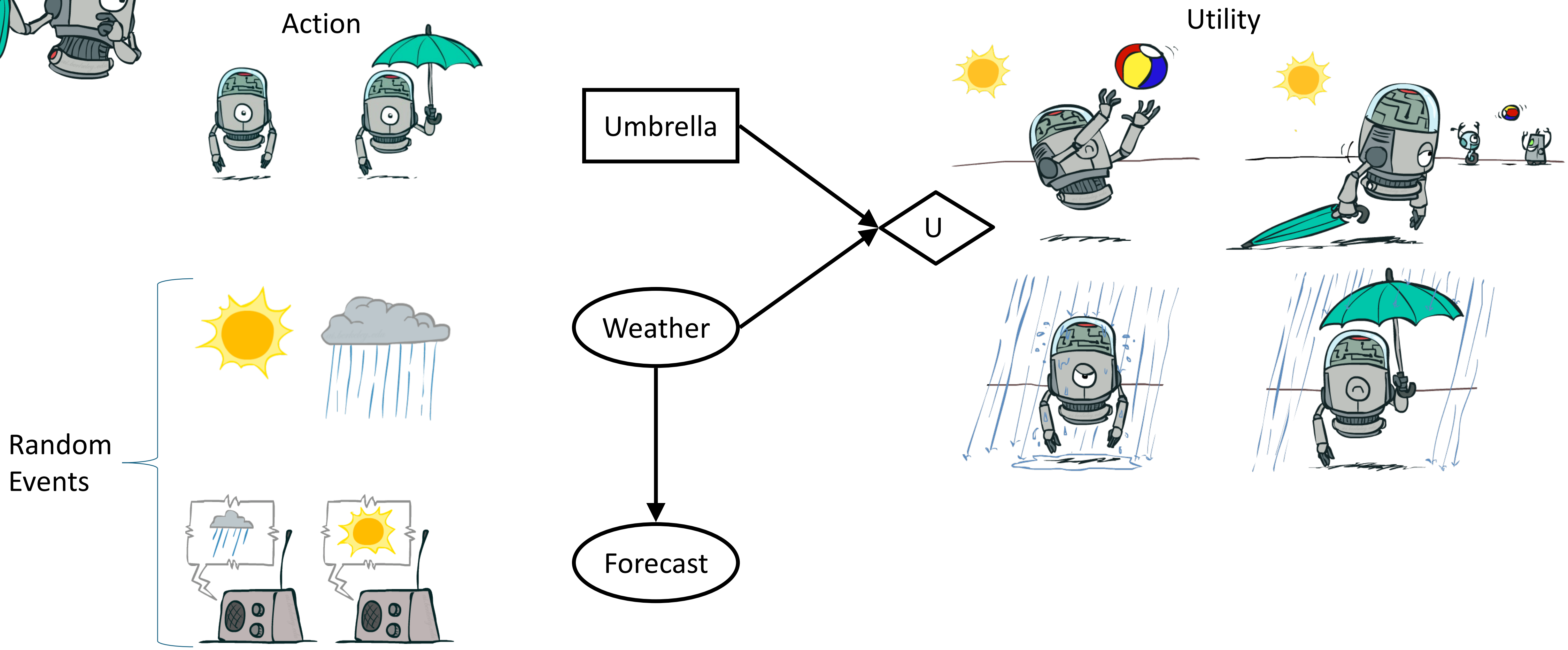
Decision Networks

Using Bayes Nets to calculate the Expected Utility of Actions.

These slides were created by Dan Klein, Pieter Abbeel, Sergey Levine, with some materials from A. Farhadi. All CS188 materials are at http://ai.berkeley.edu



Example: Decision Networks

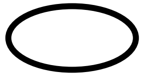


Definition: Decision Networks

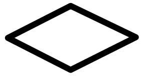
Decision networks

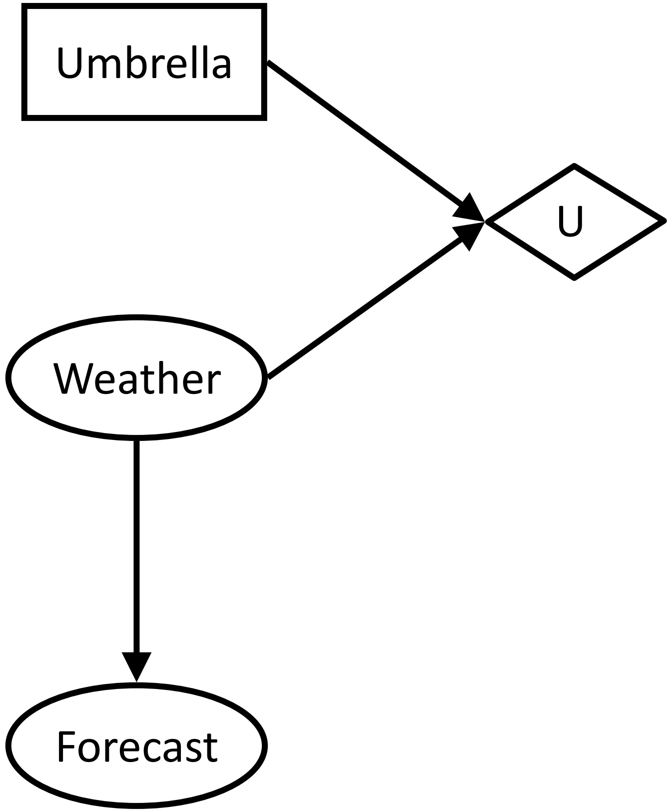
- Bayes nets with additional nodes for utility and actions.
- Allows to specify the joint probability in a compact way using independence.
- Calculate the expected utility for each possible action and choose the best.

Node types

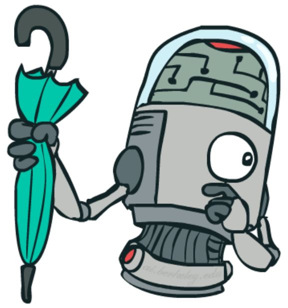

Chance nodes: Random variables in BNs


Action nodes: Cannot have parents, act as observed evidence


Utility node: Depends on action and chance nodes

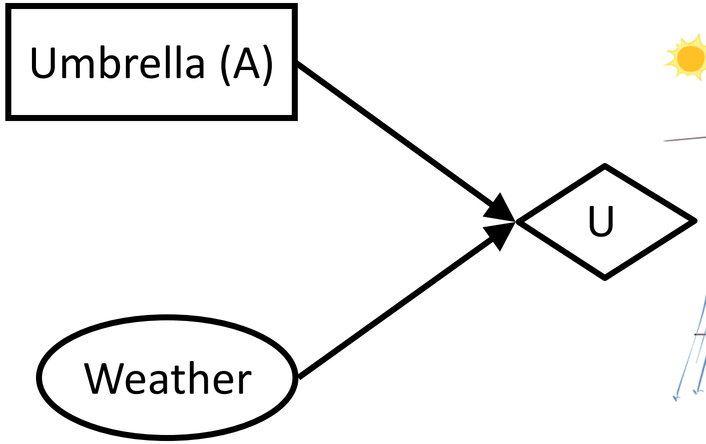


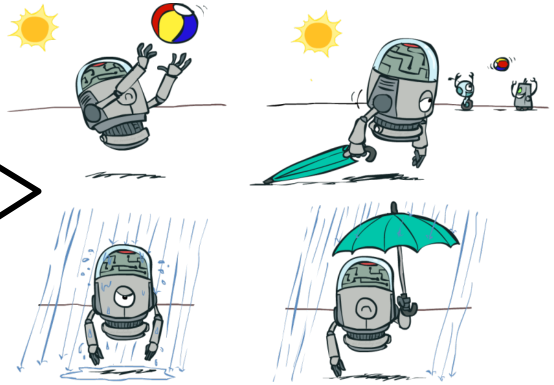
Decision Network without Forecast



Action: Umbrella = leave

$$\mathrm{EU}(\mathrm{leave}) = \sum_w P(w)U(\mathrm{leave}, w)$$
$$= 0.7 \cdot 100 + 0.3 \cdot 0 = 70$$





Action: Umbrella = take

$$\mathrm{EU}(\mathrm{take}) = \sum_w P(w)U(\mathrm{take}, w)$$
$$= 0.7 \cdot 20 + 0.3 \cdot 70 = 35$$

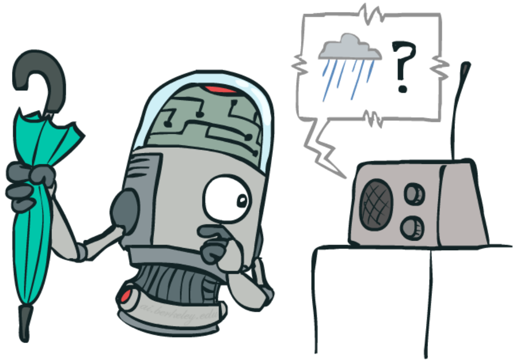
W	$P(W)$
sun	0.7
rain	0.3

A	W	$U(A,W)$
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

Optimal decision a^* = leave

$$\mathrm{MEU}(\emptyset) = \max_a \mathrm{EU}(a) = 70$$

Decision Network with Bad Forecast



Action: Umbrella = leave

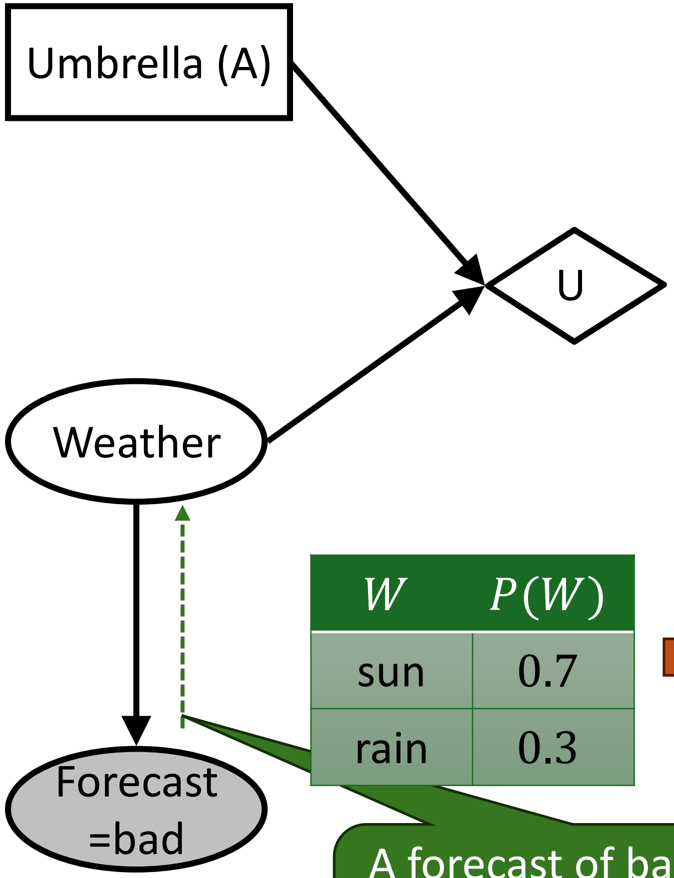
$$\mathrm{EU}(\mathrm{leave}|\mathrm{bad}) = \sum_w P(w|\mathrm{bad})U(\mathrm{leave}, w)$$
$$= 0.34 \cdot 100 + 0.66 \cdot 0 = 34$$

Action: Umbrella = take

$$\mathrm{EU}(\mathrm{take}|\mathrm{bad}) = \sum_w P(w|\mathrm{bad})U(\mathrm{take}, w)$$
$$= 0.34 \cdot 20 + 0.66 \cdot 70 = 53$$

Optimal decision a^* = leave

$$\mathrm{MEU}(F = \mathrm{bad}) = \max_a \mathrm{EU}(a|\mathrm{bad}) = 53$$



A	W	$U(A,W)$
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

W	$P(W)$
sun	0.7
rain	0.3

W	$P(W|F = bad)$
sun	0.34
rain	0.66

A forecast of bad weather is a result of an increased the probability of rain!

Conclusion

Decision networks are an extension of Bayes nets that add actions and utility to compactly specify the joint probability.

The network is used to calculate the expected utility of actions.

Decision networks can be used to make simple repeated decisions in a stochastic, partially observable, and episodic environment.

Sequential decision-making deals with decisions that influence each other and are made over time. This is a more complex decision problem and needs different methods like **Markov Decision Processes.**