# CS 188: Artificial Intelligence

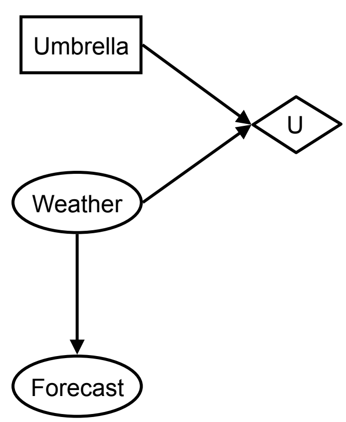
## Lecture 19: Decision Diagrams

Pieter Abbeel --- UC Berkeley

Many slides over this course adapted from Dan Klein, Stuart Russell, Andrew Moore

# Decision Networks

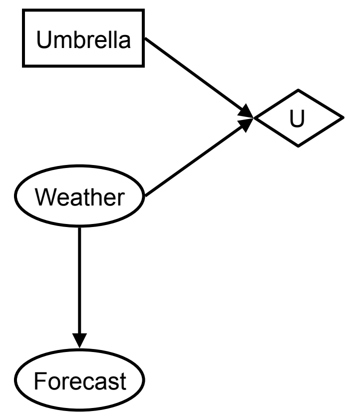
- MEU: choose the action which maximizes the expected utility given the evidence
- Can directly operationalize this with decision networks
  - Bayes nets with nodes for utility and actions
  - Lets us calculate the expected utility for each action
- New node types:
  - Chance nodes (just like BNs)
  - Actions (rectangles, cannot have parents, act as observed evidence)
  - Utility node (diamond, depends on action and chance nodes)




2

# Decision Networks

- Action selection:
  - Instantiate all evidence
  - Set action node(s) each possible way
  - Calculate posterior for all parents of utility node, given the evidence
  - Calculate expected utility for each action
  - Choose maximizing action




3

# Example: Decision Networks

Umbrella = leave

$$\mathrm{EU}(\mathrm{leave}) = \sum_{w} P(w) U(\mathrm{leave}, w)$$

$$= 0.7 \cdot 100 + 0.3 \cdot 0 = 70$$

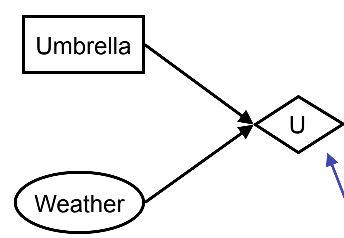
Umbrella = take

$$\mathrm{EU}(\mathrm{take}) = \sum_{w} P(w) U(\mathrm{take}, w)$$

$$= 0.7 \cdot 20 + 0.3 \cdot 70 = 35$$

Optimal decision = leave

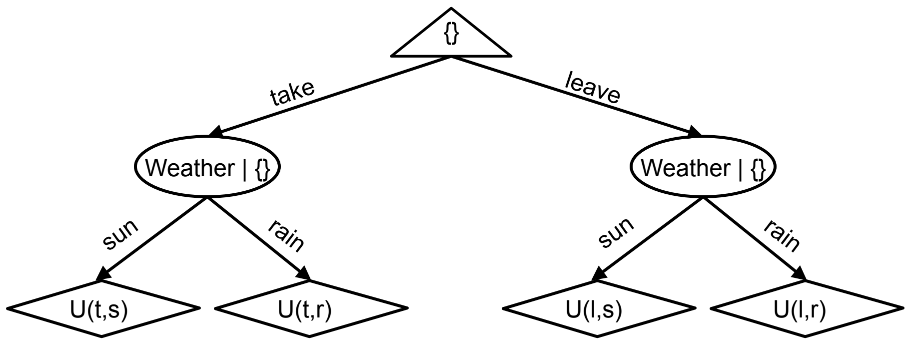
$$\mathrm{MEU}(\phi) = \max_{a} \mathrm{EU}(a) = 70$$




| 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 |

# Decisions as Outcome Trees




- Almost exactly like expectimax / MDPs
- What's changed?

5

# Example: Decision Networks

Umbrella = leave

$$\mathrm{EU}(\mathrm{leave}|\mathrm{bad}) = \sum_w P(w|\mathrm{bad})U(\mathrm{leave}, w)$$

$$= 0.34 \cdot 100 + 0.66 \cdot 0 = 34$$

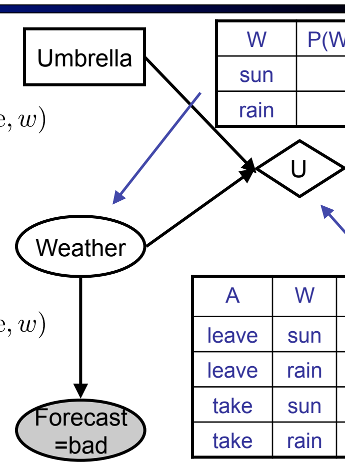
Umbrella = take

$$\mathrm{EU}(\mathrm{take}|\mathrm{bad}) = \sum_w P(w|\mathrm{bad})U(\mathrm{take}, w)$$

$$= 0.34 \cdot 20 + 0.66 \cdot 70 = 53$$

Optimal decision = take

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

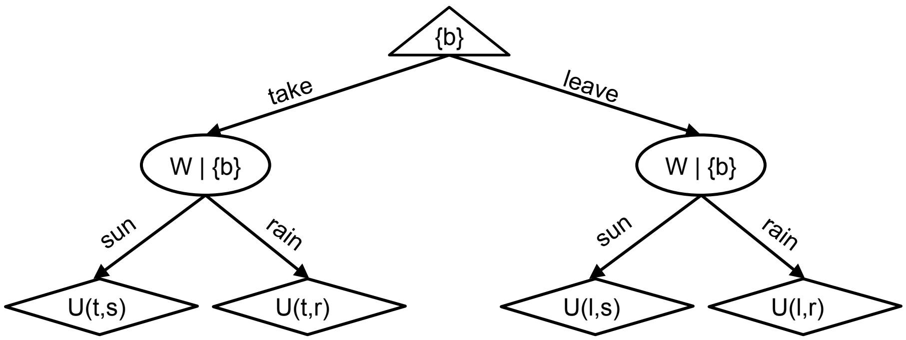



| W | P(W\|F=bad) |
|---|---|
| sun | 0.34 |
| rain | 0.66 |

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

6

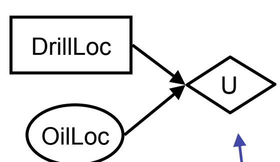
# Decisions as Outcome Trees






# Value of Information

- Idea: compute value of acquiring evidence
  - Can be done directly from decision network
- Example: buying oil drilling rights
  - Two blocks A and B, exactly one has oil, worth k
  - You can drill in one location
  - Prior probabilities 0.5 each, & mutually exclusive
  - Drilling in either A or B has EU = k/2, MEU = k/2
- Question: what's the value of information of O?
  - Value of knowing which of A or B has oil
  - Value is expected gain in MEU from new info
  - Survey may say "oil in a" or "oil in b," prob 0.5 each
  - If we know OilLoc, MEU is k (either way)
  - Gain in MEU from knowing OilLoc?
  - VPI(OilLoc) = k/2
  - Fair price of information: k/2




| O | P |
|---|---|
| a | 1/2 |
| b | 1/2 |

| D | O | U |
|---|---|---|
| a | a | k |
| a | b | 0 |
| b | a | 0 |
| b | b | k |

# VPI Example: Weather

MEU with no evidence

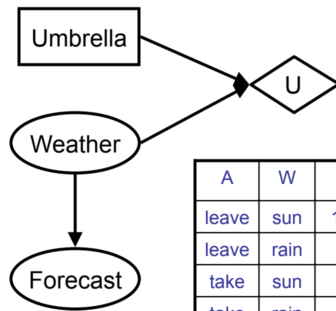
$$\text{MEU}(\emptyset) = \max_a \text{EU}(a) = 70$$

MEU if forecast is bad

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

MEU if forecast is good

$$\text{MEU}(F = \text{good}) = \max_a \text{EU}(a|\text{good}) = 95$$




| A | W | U |
|---|---|---|
| leave | sun | 100 |
| leave | rain | 0 |
| take | sun | 20 |
| take | rain | 70 |

Forecast distribution

| F | P(F) |
|---|---|
| good | 0.59 |
| bad | 0.41 |

$$0.59 \cdot (95) + 0.41 \cdot (53) - 70$$

$$77.8 - 70 = 7.8$$

$$\text{VPI}(E|e') = \left( \sum_{e'} P(e'|e)\text{MEU}(e, e') \right) - \text{MEU}(e)$$



# Value of Information

- Assume we have evidence E=e. Value if we act now:

  $$\text{MEU}(e) = \max_a \sum_s P(s|e)\, U(s, a)$$

- Assume we see that E' = e'. Value if we act then:

  $$\text{MEU}(e, e') = \max_a \sum_s P(s|e, e')\, U(s, a)$$

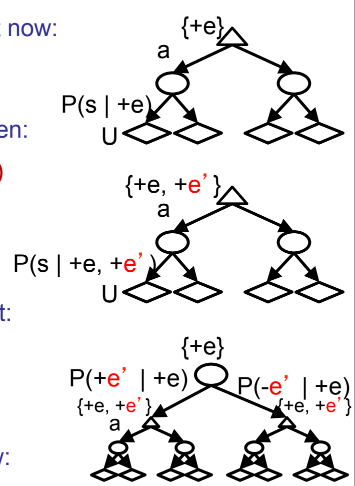
- BUT E' is a random variable whose value is unknown, so we don't know what e' will be

- Expected value if E' is revealed and then we act:

  $$\text{MEU}(e, E') = \sum_{e'} P(e'|e)\text{MEU}(e, e')$$

- Value of information: how much MEU goes up by revealing E' first then acting, over acting now:

  $$\text{VPI}(E'|e) = \text{MEU}(e, E') - \text{MEU}(e)$$



# VPI Properties

- Nonnegative

$$\forall E', e : \text{VPI}(E'|e) \geq 0$$

- Nonadditive ---consider, e.g., obtaining $E_j$ twice

$$\text{VPI}(E_j, E_k|e) \neq \text{VPI}(E_j|e) + \text{VPI}(E_k|e)$$

- Order-independent

$$\begin{aligned}\text{VPI}(E_j, E_k|e) &= \text{VPI}(E_j|e) + \text{VPI}(E_k|e, E_j) \\ &= \text{VPI}(E_k|e) + \text{VPI}(E_j|e, E_k)\end{aligned}$$

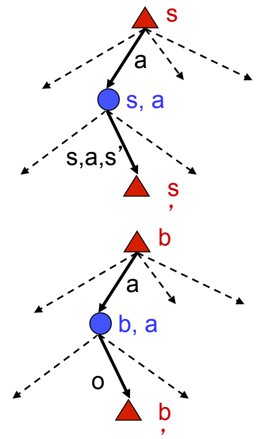


# Quick VPI Questions

- The soup of the day is either clam chowder or split pea, but you wouldn't order either one. What's the value of knowing which it is?
- There are two kinds of plastic forks at a picnic. One kind is slightly sturdier. What's the value of knowing which?
- You're playing the lottery. The prize will be $0 or $100. You can play any number between 1 and 100 (chance of winning is 1%). What is the value of knowing the winning number?

# POMDPs

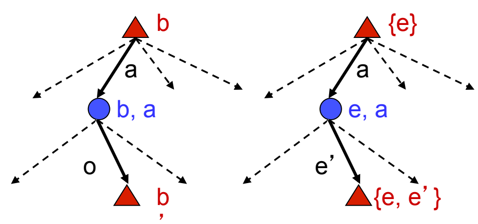
- MDPs have:
  - States S
  - Actions A
  - Transition fn P(s' |s,a) (or T(s,a,s' ))
  - Rewards R(s,a,s' )
- POMDPs add:
  - Observations O
  - Observation function P(o|s) (or O(s,o))
- POMDPs are MDPs over belief states b (distributions over S)
- We' ll be able to say more in a few lectures

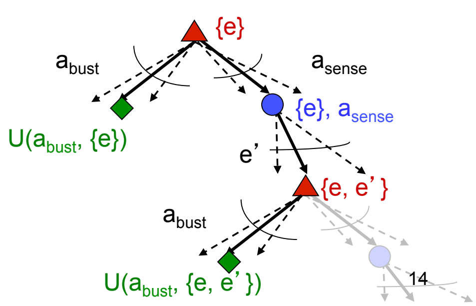





# Example: Ghostbusters

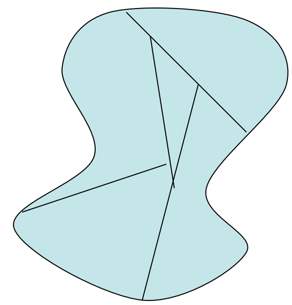
- In (static) Ghostbusters:
  - Belief state determined by evidence to date {e}
  - Tree really over evidence sets
  - Probabilistic reasoning needed to predict new evidence given past evidence
- Solving POMDPs
  - One way: use truncated expectimax to compute approximate value of actions
  - What if you only considered busting or one sense followed by a bust?
  - You get a VPI-based agent!



# More Generally

- General solutions map belief functions to actions
  - Can divide regions of belief space (set of belief functions) into policy regions (gets complex quickly)
  - Can build approximate policies using discretization methods
  - Can factor belief functions in various ways
- Overall, POMDPs are very (actually PSACE-) hard
- Most real problems are POMDPs, but we can rarely solve then in general!



15