Decision Theory: Single-Stage Decisions

CPSC 322 – Decision Theory 1

Textbook §9.1–9.2

Lecture Overview

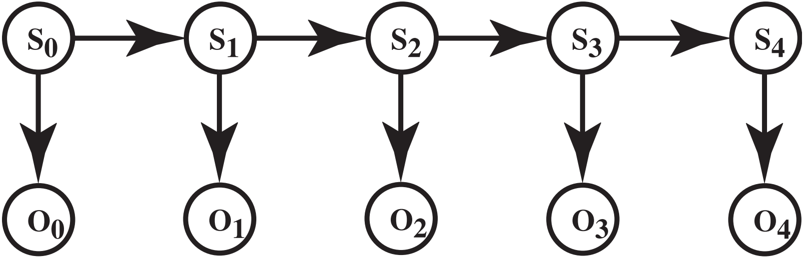
Markov chain

- A Markov chain is a special sort of belief network:

- Thus $P(S_{t+1}|S_0, \ldots, S_t) = P(S_{t+1}|S_t)$.
- Often S_t represents the state at time t. Intuitively S_t conveys all of the information about the history that can affect the future states.
- "The past is independent of the future given the present."

Hidden Markov Model

- A Hidden Markov Model (HMM) starts with a Markov chain, and adds a noisy observation about the state at each time step:



- $P(S_0)$ specifies initial conditions
- $P(S_{t+1}|S_t)$ specifies the dynamics
- $P(O_t|S_t)$ specifies the sensor model

Lecture Overview

Decisions Under Uncertainty

- In the first part of the course we focused on decision making in domains where the environment was understood with certainty
  - Search/CSPs: single-stage decisions
  - Planning: sequential decisions

- In uncertain domains, we've so far only considered how to represent and update beliefs
- What if an agent has to make decisions in a domain that involves uncertainty?
  - this is likely: one of the main reasons to represent the world probabilistically is to be able to use these beliefs as the basis for making decisions

Decisions Under Uncertainty

- An agent's decision will depend on:
  1. what actions are available
  2. what beliefs the agent has
     - note: this replaces "state" from the deterministic setting
  3. the agent's goals
- Differences between the deterministic and probabilistic settings
  - we've already seen that it makes sense to represent beliefs differently.
  - Today we'll speak about representing actions and goals
    - actions will be pretty straightforward: decision variables.
    - we'll move from all-or-nothing goals to a richer notion: rating how happy the agent is in different situations.
    - putting these together, we'll extend belief networks to make a new representation called decision networks.

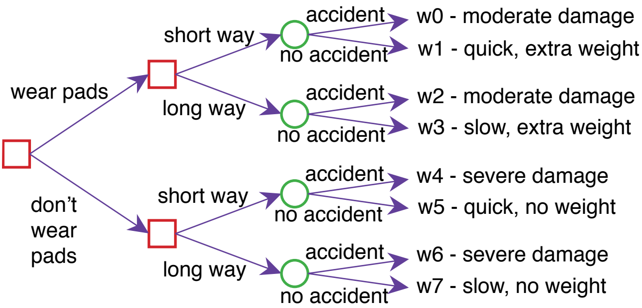
Lecture Overview

Representing Actions: Decision Variables

- Decision variables are like random variables whose value an agent gets to set.
- A possible world specifies a value for each random variable *and* each decision variable.
  - For each assignment of values to all decision variables, the measures of the worlds satisfying that assignment sum to 1.
  - The probability of a proposition is undefined unless you condition on the values of all decision variables.

Decision Tree for Delivery Robot

- The robot can choose to wear pads to protect itself or not.
- The robot can choose to go the short way past the stairs or a long way that reduces the chance of an accident.
- There is one random variable indicating whether there is an accident.



Utility

- Utility: a measure of desirability of worlds to an agent.
  - Let U be a real-valued function such that $U(\omega)$ represents an agent's degree of preference for world ω.
- Simple goals can still be specified, using a boolean utility function:
  - worlds that satisfy the goal have utility 1
  - other worlds have utility 0
- Utilities can also be more complicated. For example, in the delivery robot domain, utility might be the sum of:
  - some function of the amount of damage to a robot
  - how much energy is left
  - what goals are achieved
  - how much time it has taken.

Expected Utility

How should we define the utility of an achieving a certain probability distribution over possible worlds?

Expected Utility

How should we define the utility of an achieving a certain probability distribution over possible worlds?

- The expected value of a function of possible worlds is its average value, weighting possible worlds by their probability.
- Suppose $U(w)$ is the utility of world world w.

Definition (expected utility)

The expected utility is $\mathbb{E}(U) = \sum_{\omega \in \Omega} P(\omega) U(\omega)$.

Definition (conditional expected utility)

The conditional expected utility given e is

$$\mathbb{E}(U|e) = \sum_{\omega \models e} P(\omega|e) U(\omega).$$

Lecture Overview

Single-Stage decisions

- Given a single decision variable, the agent can choose $D = d_i$ for any $d_i \in dom(D)$.
- Write expected utility of taking decision $D = d_i$ as $\mathbb{E}(U|D = d_i)$.

Definition (optimal single-stage decision)

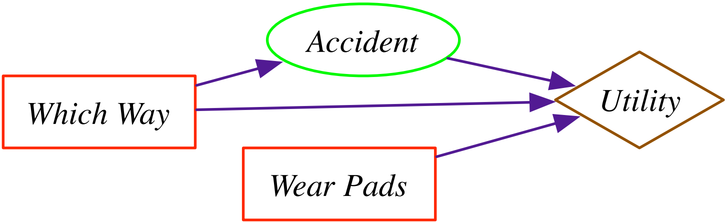
An optimal single-stage decision is the decision $D = d_{max}$ whose expected utility is maximal:

$$d_{max} \in \underset{d_i \in dom(D)}{\arg\max}\ \mathbb{E}(U|D = d_i).$$

Single-Stage-stage decision networks

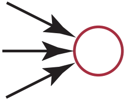
Extend belief networks with:

- Decision nodes, that the agent chooses the value for. Domain is the set of possible actions. Drawn as a rectangle.
- Utility node, the parents are the variables on which the utility depends. Drawn as a diamond.

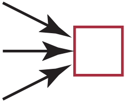


This shows explicitly which nodes affect whether there is an accident.

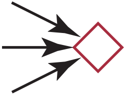
Decision Networks



- A random variable is drawn as an ellipse. Arcs into the node represent probabilistic dependence.



- A decision variable is drawn as a rectangle. Arcs into the node represent information available when the decision is made.



- A value node is drawn as a diamond. Arcs into the node represent values that the value depends on.

Finding the optimal decision

- Suppose the random variables are $X_1, \ldots, X_n$, and utility depends on $X_{i_1}, \ldots, X_{i_k}$

$$
\begin{aligned}
\mathbb{E}(U|D) &= \sum_{X_1,\ldots,X_n} P(X_1, \ldots, X_n|D) U(X_{i_1}, \ldots, X_{i_k}) \\
&= \sum_{X_1,\ldots,X_n} \prod_{i=1}^{n} P(X_i|pX_i, D) U(X_{i_1}, \ldots, X_{i_k})
\end{aligned}
$$

Finding the optimal decision

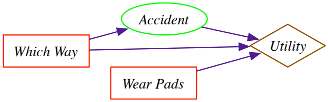
- Suppose the random variables are $X_1, \dots, X_n$, and utility depends on $X_{i_1}, \dots, X_{i_k}$

$$
\begin{aligned}
\mathbb{E}(U|D) &= \sum_{X_1,\dots,X_n} P(X_1,\dots,X_n|D)U(X_{i_1},\dots,X_{i_k}) \\
&= \sum_{X_1,\dots,X_n} \prod_{i=1}^{n} P(X_i|pX_i, D)U(X_{i_1},\dots,X_{i_k})
\end{aligned}
$$

To find the optimal decision:

- Create a factor for each conditional probability and for the utility
- Sum out all of the random variables
- This creates a factor on D that gives the expected utility for each D
- Choose the D with the maximum value in the factor.

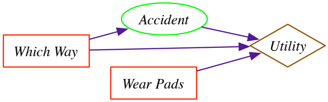
Example Initial Factors



Which Way	Accident	Probability
long	true	0.01
long	false	0.99
short	true	0.2
short	false	0.8

Which Way	Accident	Wear Pads	Utility
long	true	true	30
long	true	false	0
long	false	true	75
long	false	false	80
short	true	true	35
short	true	false	3
short	false	true	95
short	false	false	100

Example Initial Factors



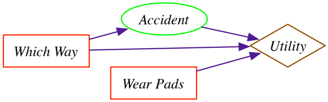
Which Way	Accident	Probability
long	true	0.01
long	false	0.99
short	true	0.2
short	false	0.8

Which Way	Accident	Wear Pads	Utility
long	true	true	30
long	true	false	0
long	false	true	75
long	false	false	80
short	true	true	35
short	true	false	3
short	false	true	95
short	false	false	100

Sum out Accident:

Which Way	Wear pads	Value
long	true	0.01*30+0.99*75=74.55
long	false	0.01*0+0.99*80=79.2
short	true	0.2*35+0.8*95=83
short	false	0.2*3+0.8*100=80.6

Example Initial Factors



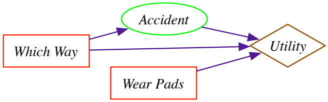
Which Way	Accident	Probability
long	true	0.01
long	false	0.99
short	true	0.2
short	false	0.8

Which Way	Accident	Wear Pads	Utility
long	true	true	30
long	true	false	0
long	false	true	75
long	false	false	80
short	true	true	35
short	true	false	3
short	false	true	95
short	false	false	100

Sum out Accident:

Which Way	Wear pads	Value
long	true	0.01*30+0.99*75=74.55
long	false	0.01*0+0.99*80=79.2
short	true	0.2*35+0.8*95=83
short	false	0.2*3+0.8*100=80.6

Example Initial Factors



Which Way	Accident	Probability
long	true	0.01
long	false	0.99
short	true	0.2
short	false	0.8

Which Way	Accident	Wear Pads	Utility
long	true	true	30
long	true	false	0
long	false	true	75
long	false	false	80
short	true	true	35
short	true	false	3
short	false	true	95
short	false	false	100

Sum out Accident:

Which Way	Wear pads	Value
long	true	0.01*30+0.99*75=74.55
long	false	0.01*0+0.99*80=79.2
short	true	0.2*35+0.8*95=83
short	false	0.2*3+0.8*100=80.6

Thus the optimal policy is to take the short way and wear pads, with an expected utility of 83.