Overview: Representation Techniques

Week 6
- Representations for classical planning problems
  - deterministic environment; complete information

Week 7
- Logic programs for problem representations
  - including planning problems, games

Week 8
- First-order logic to describe dynamic environments
  - deterministic environment; (in-)complete information

Week 9
- **State transition systems** to describe dynamic environments
  - nondeterministic environment; (in-)complete information
Decision Making

- Background: utility functions
- Decision Making in an uncertain, dynamic world

Background reading

Risk Attitudes

Which would you prefer?
- A lottery ticket that pays out $10 with probability .5 and $0 otherwise, or
- A lottery ticket that pays out $3 with probability 1

How about:
- A lottery ticket that pays out $1,000,000 with probability .5 and $0 otherwise, or
- A lottery ticket that pays out $300,000 with probability 1

- Usually, people do not simply go by expected value
- Agents are risk-neutral if they only care about the expected value
- Agents are risk-averse if they prefer the expected value to the lottery ticket
  - Most people are like this
- Agents are risk-seeking if they prefer the lottery ticket
Decreasing Marginal Utility

- Typically, at some point, having an extra dollar does not make people much happier (decreasing marginal utility)

![Graph showing decreasing marginal utility](attachment:graph.png)
Maximising Expected Utility

- Lottery 1: get $15,000 with probability 1  ⇒  expected utility = 2

- Lottery 2: get $40,000 with probability 0.4, $800 otherwise
  ⇒  expected utility = 0.4*3 + 0.6*1 = 1.8 < 2
  ⇒  expected amount of money = 0.4*$40,000 + 0.6*$800 = $16,480 > $15,000

- So: maximising expected utility is consistent with risk aversion
Acting Optimally Over Time

- **finite** number of rounds:
  Overall utility = sum of rewards (or: utility) \( u(t) \) in individual periods \( t \)

- **infinite** number of rounds:
  (Limit of) average payoff: \( \lim_{n \to \infty} \sum_{1 \leq t \leq n} u(t)/n \)
  - may not exist…
  - Discounted payoff: \( \sum_t \delta^t u(t) \) for some \( \delta < 1 \)
  
  Interpretations of discounting:
  - Interest rate
  - World ends with some probability \( 1 - \delta \)
  
  Discounting is mathematically convenient
Decision Making Under Uncertainty
Overview

- **Markov process** = state transition systems with probabilities
- Markov process + actions = **Markov decision process** (MDP)
- Markov process + partial observability = **hidden Markov model** (HMM)
- Markov process + partial observability + actions = HMM + actions = MDP with partial observability (POMDP)

<table>
<thead>
<tr>
<th></th>
<th>full observability</th>
<th>partial observability</th>
</tr>
</thead>
<tbody>
<tr>
<td>no actions</td>
<td>Markov process</td>
<td>HMM</td>
</tr>
<tr>
<td>actions</td>
<td>MDP</td>
<td>POMDP</td>
</tr>
</tbody>
</table>
Markov Processes

- time periods $t = 0, 1, 2, \ldots$
- in each period $t$, the world is in a certain state $S_t$
- **Markov assumption** – given $S_t$, $S_{t+1}$ is independent of all $S_i$ with $i < t$
  - $P(S_{t+1} \mid S_1, S_2, \ldots, S_t) = P(S_{t+1} \mid S_t)$
  - Given the current state, history tells us nothing more about the future

![Diagram of Markov Processes]

- Notation: $P(A \mid B)$ the probability of $A$ under the condition that $B$ holds
Weather Example

- $S_t$ is one of \{s, c, r\} (sun, cloudy, rain)

- Conditional transition probabilities:

  ![Diagram showing transition probabilities between states](image)

- Also need to specify an initial distribution $P(S_0)$
  - Throughout, we assume that $P(S_0 = s) = 1$
Fundamental Probability Laws

- **Law of total probability:** \( P(A) = P(A, B_1) + P(A, B_2) + P(A, B_3) \), if \( B_1, B_2, B_3 \) cover all possibilities

- **Axiom of probability:** \( P(A, B) = P(A | B) * P(B) \)

\[
P(S_{t+1} = r) = P(S_{t+1} = r, S_t = r) + P(S_{t+1} = r, S_t = s) + P(S_{t+1} = r, S_t = c)
\]

\[
P(S_{t+1} = r) = P(S_{t+1} = r | S_t = r) * P(S_t = r) + P(S_{t+1} = r | S_t = s) * P(S_t = s) + P(S_{t+1} = r | S_t = c) * P(S_t = c)
\]
Weather Example (cont'd)

What is the probability that it rains two days from now?

\[ P(S_2 = r) = P(S_2 = r, S_1 = r) + P(S_2 = r, S_1 = s) + P(S_2 = r, S_1 = c) \]

\[ = 0.1 \times 0.3 + 0.6 \times 0.1 + 0.3 \times 0.3 = 0.18 \]

What is the probability that it rains three days from now?

\[ P(S_3 = r) = P(S_3 = r | S_2 = r)P(S_2 = r) + P(S_3 = r | S_2 = s)P(S_2 = s) + P(S_3 = r | S_2 = c)P(S_2 = c) \]

\[ \Rightarrow \text{Main idea: compute distribution } P(S_1), \text{ then } P(S_2), \text{ then } P(S_3), \ldots \]
Adding Rewards to a Markov Process

- We can derive some reward from the weather each day:

![Markov Process Diagram]

- How much utility can we expect in the long run?
  - depends on the discount factor $\delta$ and the initial state

- Let $v(s)$ be the (long-term) expected utility from being in state $S$ now and $P(S,S')$ the transition probability from $S$ to $S'$

- Must satisfy $(\forall S) v(S) = u(S) + \delta \sum_{S'} P(S,S') v(S')$

  - **Example.** $v(c) = 8 + \delta(0.4v(s) + 0.3v(c) + 0.3v(r))$
  
  $\Rightarrow$ solve system of linear equations to obtain values for all states
Iteratively Updating Values

- If system of equations too had to solve because there are too many states you can iteratively update values until convergence
  - $v_i(S)$ is value estimate after $i$ iterations
  - $v_i(S) = u(S) + \delta \sum_{S'} P(S,S') v_{i-1}(S')$
- Will converge to right values
- If we initialize $v_0=0$ everywhere, then $v_i(S)$ is expected utility with only $i$ steps left (finite horizon)
Example

Let $\delta = 0.5$

- $v_0(s) = v_0(c) = v_0(r) = 0$
- $v_1(s) = 10 + 0.5 \times (0.6 \times 0 + 0.3 \times 0 + 0.1 \times 0) = 10$
  $v_1(c) = 8 + 0.5 \times (0.4 \times 0 + 0.3 \times 0 + 0.3 \times 0) = 8$
  $v_1(r) = 1 + 0.5 \times (0.2 \times 0 + 0.5 \times 0 + 0.3 \times 0) = 1$

- $v_2(s) = 10 + 0.5 \times (0.6 \times 10 + 0.3 \times 8 + 0.1 \times 1) = 14.25$
  $v_2(c) = 8 + 0.5 \times (0.4 \times 10 + 0.3 \times 8 + 0.3 \times 1) = 11.35$
  $v_2(r) = 1 + 0.5 \times (0.2 \times 10 + 0.5 \times 8 + 0.3 \times 1) = 4.15$
Markov Decision Processes
Overview

- Markov process = state transition systems with probabilities
- Markov process + actions = Markov decision process (MDP)
- Markov process + partial observability = hidden Markov model (HMM)
- Markov process + partial observability + actions = HMM + actions = MDP with partial observability (POMDP)

<table>
<thead>
<tr>
<th></th>
<th>full observability</th>
<th>partial observability</th>
</tr>
</thead>
<tbody>
<tr>
<td>no actions</td>
<td>Markov process</td>
<td>HMM</td>
</tr>
<tr>
<td>actions</td>
<td>MDP</td>
<td>POMDP</td>
</tr>
</tbody>
</table>
Markov Decision Process

- **MDP** is like a Markov process, except every round we make a decision.
- Transition probabilities depend on actions taken.
  \[ P(S_{t+1} = S' \mid S_t = s, A_t = a) = P(S, a, S') \]
- Rewards for every state, action pair.
  \[ u(S_t = s, A_t = a) \]
- Discount factor \( \delta \)

**Example.**
- A machine can be in one of three states: good, deteriorating, broken.
- Can take two actions: maintain, ignore.
Policies

- A **policy** is a function $\pi$ from states to actions

**Example**
- $\pi(\text{good shape}) = \text{ignore}$, $\pi(\text{deteriorating}) = \text{ignore}$, $\pi(\text{broken}) = \text{maintain}$

**Evaluating a policy**
- Key observation: $\text{MDP} + \text{policy} = \text{Markov process with rewards}$
- Already know how to evaluate Markov process with rewards: system of linear equations
- Algorithm for finding optimal policy: try every possible policy and evaluate
  - terribly inefficient ...
Value Iteration for Finding Optimal Policy

- Suppose you are in state $s$, and you act optimally from there on
- This leads to expected value $v^*(s)$
- Bellman equation: $v^*(s) = \max_a u(s, a) + \delta \sum_{s'} P(s, a, s') v^*(s')$

⇒ Value Iteration Algorithm

- Iteratively update values for states using Bellman equation
- $v_i(s)$ is our estimate of value of state $s$ after $i$ updates
  
  - $v_{i+1}(s) = \max_a u(s, a) + \delta \sum_{s'} P(s, a, s') v_i(s')$
  
  - If we initialize $v_0=0$ everywhere, then $v_i(s)$ is optimal expected utility with only $i$ steps left (finite horizon)

Optimal Policy

- $\pi(s) = \arg \max_a u(s, a) + \delta \sum_{s'} P(s, a, s') v^*(s')$

  take the best action
Exercise
The Monty Hall Domain

- A car prize is hidden behind one of three closed doors, goats are behind the other two
- The candidate chooses one door
- Monty Hall (the host) opens one of the other two doors to reveal a goat

- The candidate can stick to their initial choice, or switch to the other door that's still closed

Represent Monty Hall as a Markov Process with actions

- State representation: (chosen, car, open) – e.g., (3, 2, 1)

Step 1: You choose a door. Simultaneously, car is randomly placed.
Step 2: You can only do noop. Simultaneously, one door is opened.
Step 3: You can choose between noop and switch.
Markov Processes With Partial Observability
Overview

- Markov process = state transition systems with probabilities
- Markov process + actions = Markov decision process (MDP)
- Markov process + partial observability = hidden Markov model (HMM)
- Markov process + partial observability + actions = HMM + actions = MDP with partial observability (POMDP)

<table>
<thead>
<tr>
<th>full observability</th>
<th>partial observability</th>
</tr>
</thead>
<tbody>
<tr>
<td>no actions</td>
<td>Markov process</td>
</tr>
<tr>
<td>actions</td>
<td>MDP</td>
</tr>
<tr>
<td></td>
<td>HMM</td>
</tr>
<tr>
<td></td>
<td>POMDP</td>
</tr>
</tbody>
</table>
Hidden Markov Models

- Hidden Markov Model (HMM) = Markov process, but agent can't see state
- Instead, agent sees an observation each period, which depends on the current state

Transition model as before: \( P(S_{t+1} = j \mid S_t = i) = p_{ij} \)

plus observation model: \( P(O_t = k \mid S_t = i) = q_{ik} \)
HMM: Weather Example Revisited

- Observations: your labmate wet or dry
  - \(q_{sw} = 0.1, q_{cw} = 0.3, q_{rw} = 0.8\)

Example

- You have been stuck in the lab for three days (!)
- On those days, your labmate was dry, then wet, then wet again
- What is the probability that it is now raining outside?
  \[P(S_2 = r \mid O_0 = d, O_1 = w, O_2 = w)\]
  \(\Rightarrow\) Computationally efficient approach: first compute \(P(S_1 = i \mid O_0 = d, O_1 = w)\) for all states \(i\) (this is called "monitoring")
HMM: Predicting Further Out

On the last three days, your labmate was dry, wet, wet, respectively.

What is the probability that two days from now it will be raining outside?

\[ P(S_4 = r \mid O_0 = d, O_1 = w, O_2 = w) \]

Already know how to use monitoring to compute \( P(S_2 \mid O_0 = d, O_1 = w, O_2 = w) \)

\[ P(S_3 = r \mid O_0 = d, O_1 = w, O_2 = w) = \sum_S P(S_3 = r \mid S_2 = S) P(S_2 = S \mid O_0 = d, O_1 = w, O_2 = w) \]

Likewise for \( S_4 \)

⇒ So: monitoring first, then straightforward Markov process updates
Decision Making Under Partial Observability: POMDPs
Overview

- Markov process = state transition systems with probabilities
- Markov process + actions = Markov decision process (MDP)
- Markov process + partial observability = hidden Markov model (HMM)
- Markov process + partial observability + actions = HMM + actions = MDP with partial observability (POMDP)

<table>
<thead>
<tr>
<th></th>
<th>full observability</th>
<th>partial observability</th>
</tr>
</thead>
<tbody>
<tr>
<td>no actions</td>
<td>Markov process</td>
<td>HMM</td>
</tr>
<tr>
<td>actions</td>
<td>MDP</td>
<td>POMDP</td>
</tr>
</tbody>
</table>
Markov Decision Processes under Partial Observability

- POMDP = HMM + actions

Example

- Observations
  - Does machine fail on a single job?
    - \( P(\text{fail} \mid \text{good shape}) = 0.1 \)
    - \( P(\text{fail} \mid \text{deteriorating}) = 0.2 \)
    - \( P(\text{fail} \mid \text{broken}) = 0.9 \)
    - In general, probabilities can also depend on action taken
Optimal Policies in POMDPs

- Cannot simply use $\pi(s)$ because we do not know $s$
- We can maintain a probability distribution over $s$ using filtering:
  - $P(S_t | A_0 = a_0, O_0 = o_0, \ldots, A_{t-1} = a_{t-1}, O_{t-1} = o_{t-1})$
- This gives a belief state $b$ where $b(s)$ is our current probability for $s$
- Key observation: policy only needs to depend on $b$, $\pi(b)$
- If we think of the belief state as the state, then the state is observable and we have an MDP
- But: more difficult due to large, continuous state space
Exercise
Monty Hall as POMDP

Represent Monty Hall as a **Hidden** Markov Model with actions

- States representation: (chosen, car, open) – e.g., (3, 2, 1)

Step 1: You choose a door. Simultaneously, car is randomly placed (unobserved)
Step 2: You can only do noop. Simultaneously, one door is opened (observed)
Step 3: You can choose between noop and switch

What's the optimal policy?
Summary

Decision Theory
- Utility functions, discount

Single-agent decision making
- Representation: Markov Models & Hidden Markov Models
- Reasoning: MDPs & POMDPs