COMP3702/COMP7702
Artificial Intelligence

Module 3: Reasoning and planning under uncertainty — Part 1

Dr Archie Chapman

Semester 2, 2020

The University of Queensland
School of Information Technology and Electrical Engineering

Thanks to Dr Alina Bialkowski and Dr Hanna Kurniawati

Week 6: Logistics

- **Assignment 2 is due Sept 25**
- RiPPLE round 2 is due today!
- If you haven't seen them, Tutorials 5 and 6 will help you with Assignment 2

Week 7 — Overview

Continue with Module 2: Reasoning and planning with certainty

Begin Module 3: Reasoning and planning under uncertainty

- Introduction
- Probability review
- Decision theory

Table of contents

Overview of Module 3 — Reasoning and Planning under uncertainty

Causes of uncertainty: System noise and errors

Where does uncertainty come from?

- Control error or disturbances from external forces

  $\rightarrow$ Effect of performing an action is non-deterministic

- Errors in sensing and in processing of sensing data

  $\rightarrow$ Imperfect observation about the world (partially-observable)

- Strategic uncertainty and interaction with other agents, aka game theory

  $\rightarrow$ Covered in Module 5

Causes of uncertainty: Where do the modelling errors come from?

- Too complex to model

  — Lazy, e.g. rolling a dice in a casino, depends on wind direction from air conditioning, number of people around the table

  — Deliberate, to reduce computational complexity. We want to eliminate variables that will not affect the solution significantly

- Accidental error

  — e.g. lack of understanding about the problem

- Abstraction error...

Causes of uncertainty: Abstraction that may lead to modelling error

The actual possible states are often too large

Simplify, so it's solvable by current computing power:

- One approach to simplification is to clustering several actual states together and assume all actual states in the same cluster are the same
- Meaning: A state in our model corresponds to a set of actual states that are not differentiable by the program
- Similarly with action space
- Another approach is to use function approximations the state or action policy, e.g. using basis functions or machine learning methods

In both, the effect of performing an action becomes non-deterministic

Usually we deal with bounded, quantifiable uncertainty

Assumptions on environment in Module 3

- Does the agent know the state of the world/itself exactly?

  Fully observable vs partially observable
- Does an action map one state into a single other state?

  Deterministic vs non-deterministic
- Can the world change while the agent is "thinking"?

  Static vs dynamic
- Are the actions and percepts discrete

  Discrete vs continuous

Table of contents

Later: Sequential decision problems, i.e. Markov decision processes

Review of Probability

Applied probability and statistics

In this course, we are only interested in applied probability and statistics.

We will not cover the mathematics of probability theory and stochastic processes, statistics (derivations and proofs), or the design of experiments. This is not a statistics course.

If you are unfamiliar with probability or want more detail, see:

- R&N Chapter 12.1–12.5
- P&M Chapter 8.1

Some terminology

Experiment: An occurrence with an uncertain outcome that we can observe.

For example, rolling a die.

Outcome: The result of an experiment; one particular state of the world.

For example: 4.

Sample Space: The set of all possible outcomes for the experiment.

For example, 1, 2, 3, 4, 5, 6.

Event: A subset of possible outcomes that together have some property we are interested in.

For example, the event "even die roll" is the set of outcomes 2, 4, 6.

Probability...

What is a probability distribution?

What do you think of when an event is described as "random"?

- An unexpected event?
- A uniform number between 0 and 1?
- A normally distributed random value?

A *random variable*, denoted X, has an element of chance associated with its value.

The level of chance associated with any particular value (or range of values), $X = x$, is called its **probability**, $P(X = x)$. This is a positive value between 0 and 1.

The collection of probabilites over values that a variable may take is called a **distribution**, with the property that the sum of probabilities of all mutually exclusive, collectively exhaustive events is 1.

The value of any function of a random varible is also a random variable. E.g. the sum of n random variables takes a random value.

What is a probability distribution?

Very loosely* - just about anything that you can count or measure, has non-negative values, and that sums to one over all outcomes, is a probability distribution.

Fundamentally, both discrete and continuous variables, X, are represented by a *cumulative distribution function*, cdf, denoted $F(x)$.

The cdf is the probability that the realised value of X is less than or equal to x:

$$F(x) = P(X \leq x).$$

(**Take it on trust that there is a serious branch of mathematics behind this, regarding topological spaces and measure theory.*)

What is a probability distribution?

The terms used for the probability that X takes a particular value are different for discrete and continuous variables.

For discrete variables, a *probability mass function* (pmf), $P(X = x)$ describes the chance of a particular event occurring.

- For finite discrete-valued variables, this is easy to understand as a finite vector of non-negative values that sum to one. E.g. chance of a coin toss, roll of dice, poker hands, etc.
- For countably infinite discrete variables, the probability distribution is a series of numbers over an infinite set of distinct elements.

For continuous variables, a *probability density function* (pdf), $f(x)$, is a continuous function that integrates from below to the cumulative density function:

$$F(X \leq x) = \int_{-\infty}^{x} f(y).dy$$

Probability distributions

Python will very effectively help you handle probabilities.

Many distributions have functions for probabilities, random number generation, etc.

The methods that you will learn about available through the python `random` module, which is part of the standard library. See `https://docs.python.org/3/library/random.html` for details.

Sampling random variables

For example, a *normal distribution* can be sampled using:

```
>> import random as r
>> mu, sigma = 2.0, 4.0
>> x = r.gauss(mu, sigma)
1.3927394833370967
```

...or a Weibull distribution, $f(x|a, b) = \frac{b}{a}\left(\frac{x}{a}\right) e^{-\left(\frac{x}{a}\right)^k}$:

```
>> a, b = 1.0, 1.5
>> r.weibullvariate(a, b)
1.9157188803236334
```

In this course, you don't have to remember functional forms, etc — just focus on understanding a distribution's use and parameters.

Random numbers: discrete uniform distribution and choices from sets

Even integer from 0 to 100 inclusive:

```
>> randrange(0, 101, 2)
26
```

Single random element from a sequence

```
>> options =['win', 'lose', 'draw']
>> choice(options)
'draw'
```

Single element according to relative weights

```
>> weights = [5,10,15]
>> r.choices(options,weights)
'lose'
```

Conditional probability and independence

Conditional probability is a measure of the probability of an event given that another event has already occurred.

If the event of interest is A and the event B is known to have occurred, then the corresponding conditional probability of A given B is denoted $P(A \mid B)$.

If two events, A and B, are independent, then the probability of both occurring is

$$P(A \cap B) = P(A)\,P(B)$$

Otherwise, if the events are dependent:

$$P(A \cap B) = P(B)\,P(A \mid B) = P(A)\,P(B \mid A)$$

In both cases, the probability of events A or B occuring is:

$$P(A \cup B) = P(A) + P(B) - P(B \cap A)$$

(Nb. the version of the slide used in the lecture included the incorrect statement for independent events: $P(A \cup B) = P(A) + P(B)$. This is not true for independent events; instead it applies only to *mutually exclusive* events.)

Bayes rule

Bayes' rule rearranges the conditional probability relationships to describe the probability of an event, given prior knowledge of related events:

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

E.g. Knowing symptoms of a disease is easier than figuring out the disease given the symptoms.

Search under uncertainty: AND-OR trees

Making decisions

- We want to find a plan that works regardless of what outcomes actually occur
- Can no longer rely on a sequence of actions
- Need a conditional plan. The action to perform depends on the output of the previous action
- Need a different type of tree data structure

AND-OR search tree

- A tree with interleaving AND and OR levels
- At each node of an OR level, branching is introduced by the agent's own choice
- At each node of an AND level, branching is introduced by the environment

Example: Slippery vacuum robot

States: Conjunctions of the following state factors:

- Robot Position: $\{\text{in } R_1, \text{ in } R_2\}$
- R_1 state: {clean, dirty}
- R_2 state: {clean, dirty}

Action: { Left, Right, Suck(R_1), Suck(R_2) }

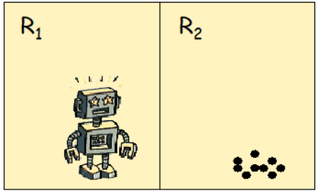
World dynamics: Non deterministic, after performing an action at a state, the robot may end up in one of several possible states

Initial state:

- (Robot in R_1) $\wedge$ (R_1 is clean) $\wedge$ (R_2 is dirty)

Goal state:

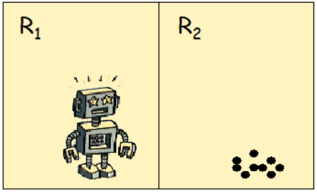
- (R_1 is clean) $\wedge$ (R_2 is clean)



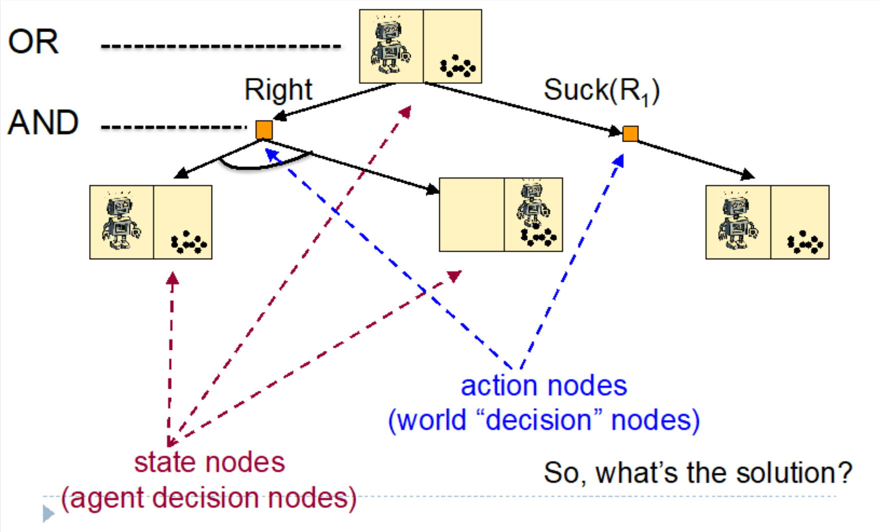
Example: Slippery vacuum robot

World dynamics:

- Successors of (Robot in R_1, Right) = {Robot in R_1, Robot in R_2}
- Successors of (Robot in R_2, Right) = {Robot in R_1, Robot in R_2}



AND-OR tree of the slippery vacuum robot



So, what's the solution?

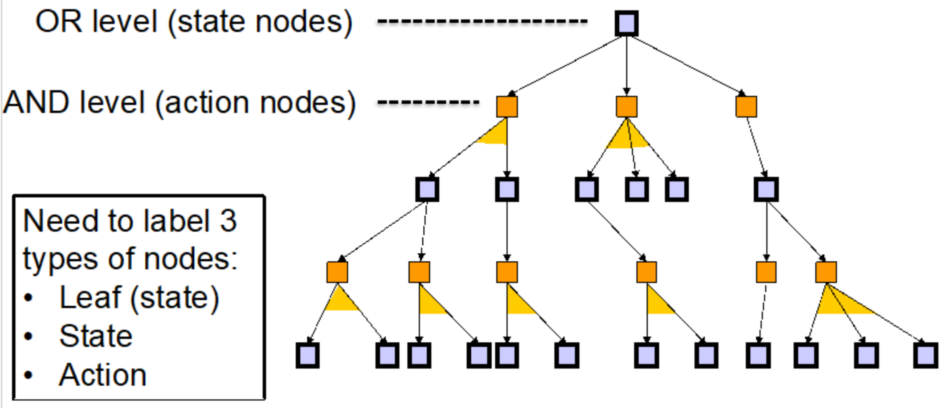
AND-OR search tree

Solution is a sub-tree that:

- Has a goal node at every leaf
- Specifies one action at each node of an OR level
- Includes every outcome branch at each node of an AND level

When do we have a solution?

Labelling an AND-OR tree

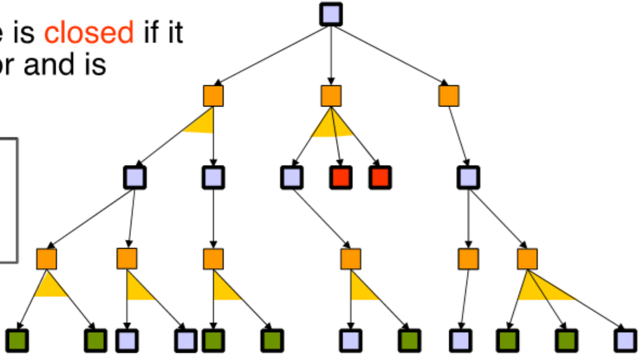


Labelling an AND-OR tree

- A leaf state node is solved if it's a goal state
- A leaf state node is closed if it has no successor and is not a goal

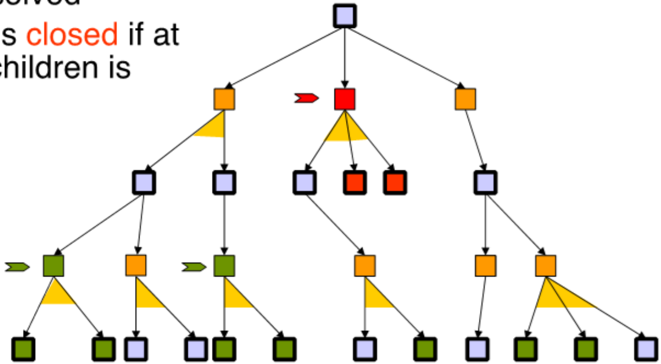
Can view:

- Solved as True
- Closed as False



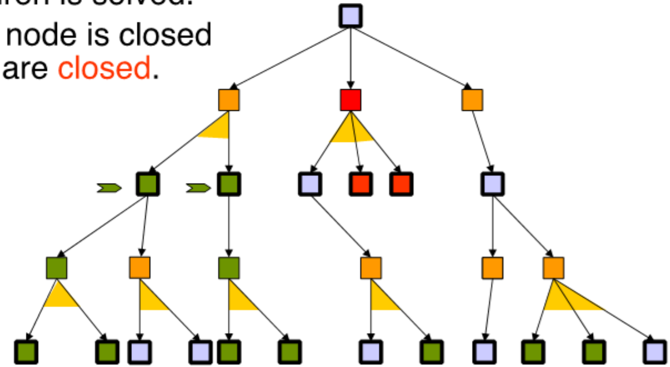
Labelling an AND-OR tree

- An action node is solved if all its children are solved
- An action node is closed if at least one of its children is closed



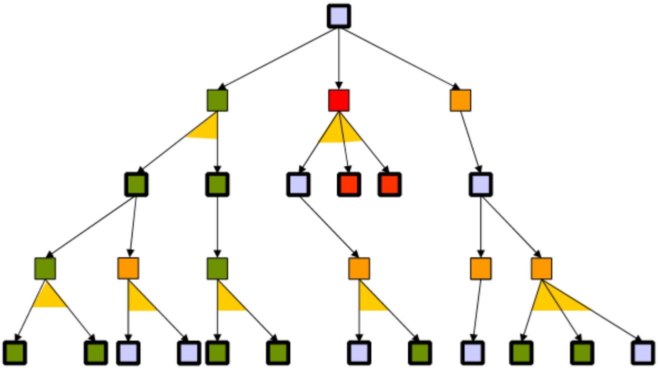
Labelling an AND-OR tree

- A non-leaf state node is solved if one of its children is solved.
- A non-leaf state node is closed if all its children are closed.



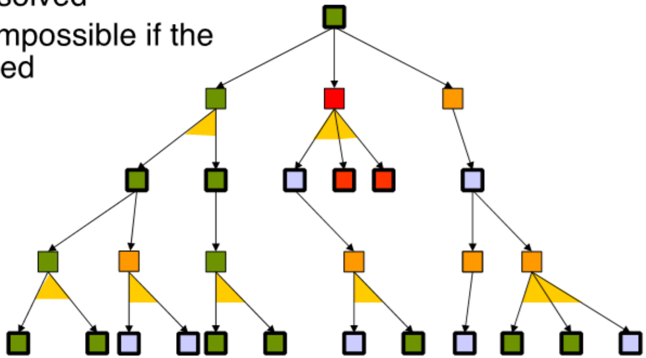
Labelling an AND-OR tree

- Keep labeling until the root.



Labelling an AND-OR tree

- The problem is solved when the root node is solved
- The problem is impossible if the root node is closed



When a node is the same as an ancestor node. . . ?

Search AND-OR search tree

Start from a state node (OR level)

- Fringe nodes are state nodes

Use any of the search algorithms we have studied,

- Select a fringe node to expand
- Select an action to use
- Insert the corresponding action node
- Insert all possible outcomes of the action, as the child of the action node
- Backup to (re-)label the ancestor nodes

Cost/reward calculation at AND level:

- Weighted sum (when uncertainty is quantified using probability, expectation)
- Take the maximum cost / minimum reward (conservative)

Decision theory

Preferences

- Actions result in outcomes
- Agents have **preferences** over outcomes
- A rational agent will do the action that has the best outcome for them
- Sometimes agents don't know the outcomes of the actions, but they still need to compare actions
- Agents have to act.
  (Doing nothing is (often) an action).

Preferences Over Outcomes

Some notation:

- The preference relation, $\succ$, means "is preferred to" or "succeeds in a preference order"
- $\prec$ is "precedes in a preference order"
- Indifference is $\sim$

If o_1 and o_2 are outcomes

- $o_1 \succeq o_2$ means o_1 is at least as desirable as o_2.
- $o_1 \sim o_2$ means $o_1 \succeq o_2$ and $o_2 \succeq o_1$.
- $o_1 \succ o_2$ means $o_1 \succeq o_2$ and $o_2 \not\succeq o_1$

Lotteries

- An agent may not know the outcomes of its actions, but only have a probability distribution of the outcomes.
- o_1 lottery is a probability distribution over outcomes.
- R&N denote this $[p_1, o_1; p_2, o_2, \ldots, p_k, k]$, and
  P&M denote it: $[p_1 : o_1, p_2 : o_2, \ldots, p_k : k]$ (like a python dict)
  where the i are outcomes and $p_i \geq 0$ such that $\sum_i p_i = 1$
- The lottery specifies that outcome i occurs with probability p_i.
- When we talk about outcomes, we will include lotteries over "pure" outcomes.

Axioms of rational preferences

Idea: preferences of a rational agent must obey certain rules.

Rational preferences imply behaviour describable as maximization of expected utility

Completeness (for some reason R&N call this *Orderability*): $(o_1 \succ o_2) \vee (o_2 \prec o_1) \vee (o_1 \sim o_2)$

Transitivity: $(o_1 \succ o_2) \wedge (o_2 \succ C) \Rightarrow (o_1 \succ C)$

Monotonicity: $o_1 \succ o_2 \Rightarrow (p \geq q \Leftrightarrow [p : o_1, 1-p : o_2] \succeq [q : o_1, 1-q : o_2])$

Continuity: $o_1 \succ o_2 \succ C \Rightarrow \exists p \in [0,1][p : o_1, 1-p : C] \sim o_2$

Substitutability: $o_1 \sim o_2 \Rightarrow [p : o_1, 1-p : C] \sim [p : o_2, 1-p : C]$

Decomposability $[p : o_1, 1-p : [q : o_2, 1-q : o_3]] \sim [p : o_1, (1-p)q : o_2, (1-p)(1-q)o_3]$

Properties of Preferences — Completeness

Completeness: Agents have to act, so they must have preferences:

$$\forall o_1 \forall o_2 \;\; o_1 \succeq o_2 \text{ or } o_2 \succeq o_1$$

Properties of Preferences — Transitivity

Transitivity: Preferences must be transitive:

$$\text{if } o_1 \succeq o_2 \text{ and } o_2 \succ o_3 \text{ then } o_1 \succ o_3$$

(Similarly for other mixtures of $\succ$ and $\succeq$.)

Rationale: otherwise $o_1 \succeq o_2$ and $o_2 \succ o_3$ and $o_3 \succeq o_1$.
If they are prepared to pay to get o_2 instead of o_3,
and are happy to have o_1 instead of o_2,
and are happy to have o_3 instead of o_1
$\longrightarrow$ money pump.

Properties of Preferences — Monotonicity

Monotonicity: An agent prefers a larger chance of getting a better outcome than a smaller chance:

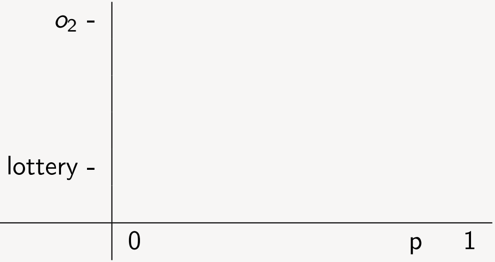
- If $o_1 \succ o_2$ and $p > q$ then

$$[p : o_1, 1 - p : o_2] \succ [q : o_1, 1 - q : o_2]$$

Consequence of axioms of Completeness, Transitivity and Monotonicity

- Suppose $o_1 \succ o_2$ and $o_2 \succ o_3$. Consider whether the agent would prefer
  - o_2
  - the lottery $[p : o_1, 1 - p : o_3]$

  for different values of $p \in [0, 1]$.
- Plot which one is preferred as a function of p:



Properties of Preferences — Continuity

Continuity: Suppose $o_1 \succ o_2$ and $o_2 \succ o_3$, then there exists a $p \in [0, 1]$ such that

$$o_2 \sim [p : o_1, 1 - p : o_3]$$

Properties of Preferences — Substitutability

Substitutability: if $o_1 \sim o_2$ then the agent is indifferent between lotteries that only differ by o_1 and o_2:

$$[p : o_1, 1 - p : o_3] \sim [p : o_2, 1 - p : o_3]$$

Alternative Axiom for Substitutability

Substitutability: if $o_1 \succeq o_2$ then the agent weakly prefers lotteries that contain o_1 instead of o_2, everything else being equal.

That is, for any number p and outcome o_3:

$$[p : o_1, (1-p) : o_3] \succeq [p : o_2, (1-p) : o_3]$$

Properties of Preferences — Decomposability

Decomposability: (no fun in gambling). An agent is indifferent between lotteries that have same probabilities and outcomes. This includes lotteries over lotteries.

For example:

$$\begin{aligned} &[p : o_1, 1 - p : [q : o_2, 1 - q : o_3]] \\ \sim \quad &[p : o_1, (1 - p)q : o_2, (1 - p)(1 - q) : o_3] \end{aligned}$$

Summary of Rational Preferences

Completeness:

$$(o_1 \succ o_2) \vee (o_2 \prec o_1) \vee (o_1 \sim o_2)$$

Transitivity:

$$(o_1 \succ o_2) \wedge (o_2 \succ C) \Rightarrow (o_1 \succ C)$$

Monotonicity:

$$o_1 \succ o_2 \Rightarrow (p \geq q \Leftrightarrow [p : o_1, 1 - p : o_2] \succeq [q : o_1, 1 - q : o_2])$$

Continuity:

$$o_1 \succ o_2 \succ C \Rightarrow \exists p \in [0, 1][p : o_1, 1 - p : C] \sim o_2$$

Substitutability:

$$o_1 \sim o_2 \Rightarrow [p : o_1, 1 - p : C] \sim [p : o_2, 1 - p : C]$$

Decomposability $[p : o_1, 1 - p : [q : o_2, 1 - q : o_3]] \sim [p : o_1, (1 - p)q : o_2, (1 - p)(1 - q)o_3]$

What we would like

- We would like a measure of preference that can be combined with probabilities. So that

$$\begin{aligned} & value([p : o_1, 1 - p : o_2]) \\ = \quad & p \times value(o_1) + (1 - p) \times value(o_2) \end{aligned}$$

- Money does not act like this.
  What would you prefer

$$\$1,000,000 \text{ or } [0.5 : \$0, 0.5 : \$2,000,000]?$$

- It may seem that preferences are too complex and muti-faceted to be represented by single numbers.

Theorem

If preferences follow the preceding properties, then preferences can be measured by a function

$$utility : outcomes \rightarrow [0, 1]$$

such that

- $o_1 \succ o_2$ if and only if $utility(o_1) \geq utility(o2)$
- $o_1 \sim o_2$ if and only if $utility(o_1) = utility(o2)$
- Utilities are linear with probabilities:

$$\begin{aligned} & utility([p_1 : o_1, p_2 : o_2, \ldots, p_k : o_k]) \\ = \quad & \sum_{i=1}^{k} p_i \times utility(o_i) \end{aligned}$$

So, we can replace preferences with real-numbers, but still what is the "best" lottery?

Maximum Expected Utility (MEU)

Main problem: What does "best" mean?

Utility: a number that assigns the desirability of a state, MEU is the commonly used definition of "best" decision

Idea:

- Assigns utility function to each outcome (state) to represent the agent's preference
- "Best" decision maximises the expected utility of the outcomes

Example: Buying a Used Car

Goal of buying the car: To gain profit from reselling it

Car costs \$1000

Can sell the car for \$1100 $\Rightarrow$ \$100 profit

BUT Every car is either good or bad

- Costs \$40 to repair a good car
- Costs \$200 to repair a bad car
- 20% cars are bad

Should we buy the car? $\rightarrow$ Solve using MEU

Example: Buying a Used Car

State space: $\{$good car, bad car$\}$

Preference: good car $\succ$ bad car

Utility function:

- U(good car) =1100 - 1000 - 40 = 60
- U(bad car) = 1100 - 1000 - 200 = -100

Lottery: [0.8, good car; 0.2, bad car]

Expected Utility if we buy the car:

- $P(goodcar) \times U(goodcar) + P(badcar) \times U(badcar) 0.8 \times 60 + 0.2 \times -100 = 28$
- Higher than not buying the car. Hence, **buy!**

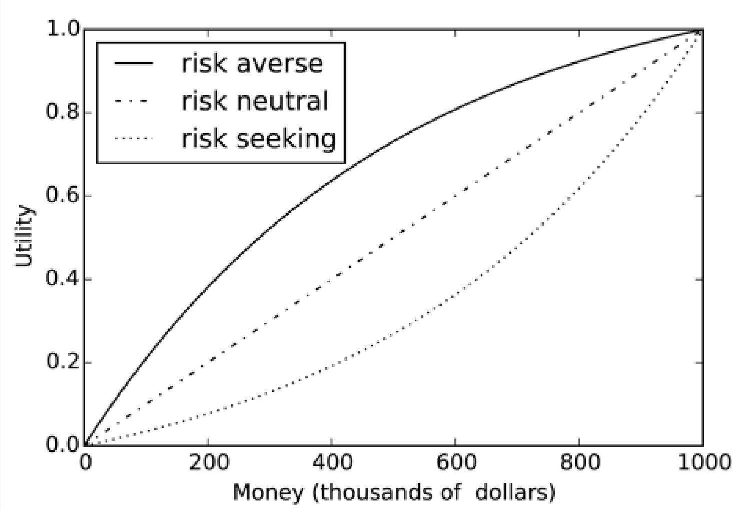
How about the Utility of Money?

Which one do you prefer?
A: A sure gain of \$240 **B**: A 25% chance of winning \$1000 and 75% chance of winning nothing

Which one do you prefer? **C**: A sure loss of \$750 **D**: A 75% chance of losing \$1000 and 25% chance of losing nothing

Is decision theory useless here? Need a better utility function that can incorporate our preference! (must follow the axioms)

Utility as a function of money



Factored Representation of Utility

- Suppose the outcomes can be described in terms of features $X_1, \ldots, X_n$.
- An additive utility is one that can be decomposed into set of factors:

$$u(X_1, \ldots, X_n) = f_1(X_1) + \cdots + f_n(X_n).$$

  This assumes additive independence.
- Strong assumption: contribution of each feature doesn't depend on other features.
- Many ways to represent the same utility:
  — a number can be added to one factor as long as it is subtracted from others.

Additive Utility

- An additive utility has a canonical representation:

$$u(X_1, \ldots, X_n) = w_1 \times u_1(X_1) + \cdots + w_n \times u_n(X_n).$$

- If $best_i$ is the best value of X_i, $u_i(X_i{=}best_i) = 1$.
  If $worst_i$ is the worst value of X_i, $u_i(X_i{=}worst_i) = 0$.
- w_i are weights, $\sum_i w_i = 1$.
  The weights reflect the relative importance of features.
- We can determine weights by comparing outcomes.

$$w_1 = u(best_1, x_2, \ldots, x_n) - u(worst_1, x_2, \ldots, x_n).$$

  for any values $x_2, \ldots, x_n$ of $X_2, \ldots, X_n$.

Complements and Substitutes

- Often additive independence is not a good assumption.
- Values x_1 of feature X_1 and x_2 of feature X_2 are complements if having both is better than the sum of the two.
- Values x_1 of feature X_1 and x_2 of feature X_2 are substitutes if having both is worse than the sum of the two.
- Example: on a holiday
  - An excursion for 6 hours North on day 3.
  - An excursion for 6 hours South on day 3.
- Example: on a holiday
  - A trip to a location 3 hours North on day 3
  - The return trip for the same day.

Generalized Additive Utility

- A generalized additive utility can be written as a sum of factors:

$$u(X_1, \ldots, X_n) = f_1(\overline{X_1}) + \cdots + f_k(\overline{X_k})$$

  where $\overline{X_i} \subseteq \{X_1, \ldots, X_n\}$.
- An intuitive canonical representation is difficult to find.
- It can represent complements and substitutes.

Utility and time

- Would you prefer \$1000 today or \$1000 next year?
- What price would you pay now to have an eternity of happiness?
- How can you trade off pleasures today with pleasures in the future?

Utility and time

- How would you compare the following sequences of rewards (per week):

  A: \$1000000, \$0, \$0, \$0, \$0, \$0,...
  B: \$1000, \$1000, \$1000, \$1000, \$1000,...
  C: \$1000, \$0, \$0, \$0, \$0,...
  D: \$1, \$1, \$1, \$1, \$1,...
  E: \$1, \$2, \$3, \$4, \$5,...

Rewards and Values

Suppose the agent receives a sequence of rewards $r_1, r_2, r_3, r_4, \ldots$ in time. What utility should be assigned? "Return" or "value"

- total reward $V = \sum_{i=1}^{\infty} r_i$
- average reward $V = \lim_{n\to\infty} (r_1 + \cdots + r_n)/n$

Suppose the agent receives a sequence of rewards $r_1, r_2, r_3, r_4, \dots$ in time.

- discounted return $V = r_1 + \gamma r_2 + \gamma^2 r_3 + \gamma^3 r_4 + \cdots$
  γ is the discount factor $0 \le \gamma \le 1$.

Properties of the Discounted Rewards

- The discounted return for rewards $r_1, r_2, r_3, r_4, \ldots$ is

$$\begin{aligned} V &= r_1 + \gamma r_2 + \gamma^2 r_3 + \gamma^3 r_4 + \cdots \\ &= r_1 + \gamma(r_2 + \gamma(r_3 + \gamma(r_4 + \ldots))) \end{aligned}$$

- If V_t is the value obtained from time step t

$$V_t = r_t + \gamma V_{t+1}$$

- How is the infinite future valued compared to immediate rewards?
  $1 + \gamma + \gamma^2 + \gamma^3 + \cdots = 1/(1-\gamma)$
  Therefore $\dfrac{\text{minimum reward}}{1-\gamma} \leq V_t \leq \dfrac{\text{maximum reward}}{1-\gamma}$
- We can approximate V with the first k terms, with error:

$$V - (r_1 + \gamma r_2 + \cdots + \gamma^{k-1} r_k) = \gamma^k V_{k+1}$$

Allais Paradox (1953)

What would you prefer:

A: $\$1m$ — one million dollars

B: lottery $[0.10 : \$2.5m, 0.89 : \$1m, 0.01 : \$0]$

What would you prefer:

C: lottery $[0.11 : \$1m, 0.89 : \$0]$

D: lottery $[0.10 : \$2.5m, 0.9 : \$0]$

It is inconsistent with the axioms of preferences to have $A \succ B$ and $D \succ C$.

A,C: lottery $[0.11 : \$1m, 0.89 : X]$

B,D: lottery $[0.10 : \$2.5m, 0.01 : \$0, 0.89 : X]$

Framing Effects [Tversky and Kahneman]

- A disease is expected to kill 600 people. Two alternative programs have been proposed:

  Program A: 200 people will be saved
  Program B: probability 1/3: 600 people will be saved
  probability 2/3: no one will be saved

  Which program would you favor?

- A disease is expected to kill 600 people. Two alternative programs have been proposed:

  Program C: 400 people will die
  Program D: probability 1/3: no one will die
  probability 2/3: 600 will die

  Which program would you favor?

Tversky and Kahneman: 72% chose A over B.

22% chose C over D.

Attributions and References

Thanks to Dr Alina Bialkowski and Dr Hanna Kurniawati for their materials.

Other materials derived from Stuart Russell and Peter Norvig, *Artificial Intelligence: A Modern Approach*, 3E, Prentice Hall, 2009.

All remaining errors are Archie's — please email if you find any: archie.chapman@uq.edu.au