Representing Utilities

The same issues arise for utility as for probability.

Suppose that preference is represented in terms of a utility function. How hard is it to describe the function?

- If an outcome depends on n factors, each with at least k possible values, get at least k^n possible outcomes.
  - Describing the utility function can be hard!

Example: Consider buying a house. What matters?

- price of house (p)
- distance from school (ds)
- quality of local school (sq)
- distance from work (dw)
- condition of house (c)

Thus, utility is a function of these 5 parameters (and maybe others):

$$u(p, ds, sq, dw, c)$$

Suppose each parameter has three possible values.

- Describing the utility function seems to require $3^5 = 243$ numbers.

We can do better if the utility is additively separable:

$$u(p, ds, sq, dw, c) = u_1(p)+u_2(ds)+u_3(sq)+u_4(dw)+u_5(c)$$

There are only 15 numbers to worry about

- We compute $u_1, \ldots, u_5$ separately

With additive separability, can consider each attribute independently.

- Seems reasonable in the case of the house.

Additive separability doesn't always hold. We want

- General conditions that allow for simpler descriptions of utilities
- Graphical representations that allow for easier representation
- Techniques to make utility elicitation easier

Formalities

- $V = \{V_1, \ldots, V_n\}$ is a set of factors (attributes)
- Each V_i takes on possible values in some set D_i.
- Let O consist of all tuples in $D_1 \times \ldots \times D_n$: the outcome space.
  - An outcome describes the values of all factors
- An *O-lottery* is a probability distribution over O
  - describes how likely each outcome in O is
- A utility u on O induces a preference ordering $\preceq_u$ on O-lotteries:
  - $p_1 \preceq_u p_2$ iff $\Sigma_{o \in O} p_1(o)u(o) \leq \Sigma_{o \in O} p_2(o)u(o)$
    - $*$ expected utility of $p_1 \leq$ expected utility of p_2

Utility Independence

- If $X \subseteq V$, let $O_X = \Pi_{V_i \in X} D_i$
- $o' \in O_X$ corresponds to a subset $S(o')$ of O
  - $S(o') = \{o \in O \mid o_X = o'\}$
- Given an O-lottery p, let p_X be the marginal on O_X:
$$p_X(o') = p(O(o'))$$
- Given an O_X-lottery p and $o' \in O_{V-X}$, let $p^{o'}$ be the O-lottery such that $p^{o'}_X = p$, $p^{o'}(O(o')) = 1$
- Define ordering $\preceq_u^{o'}$ on O_X-lotteries:
$$p \preceq_u^{o'} q \text{ iff } p^{o'} \preceq_u q^{o'}$$

Definition: X is *utility independent* of $V - X$ (w.r.t $\preceq_u$) if preferences for lotteries over X do not depend on the value of factors in $V - X$: for all O_X-lotteries p, q and all $o, o' \in O_{V-X}$,

$$p^o \preceq_u q^o \text{ iff } p^{o'} \preceq_u q^{o'}$$

Theorem 1: Every subset $X \subseteq V$ is utility independent of $V - X$ iff there is either an additive or a multiplicative representation of utility; i.e. there exist functions f_i on D_i such that either

(a) $\exists c \, \forall (v_1, \ldots, v_n) \in O \, [u(v_1, \ldots, v_n) = \Pi_{i=1}^n f_i(v_i) + c]$

(b) $\forall (v_1, \ldots, v_n) \in O \, [u(v_1, \ldots \ldots, v_n) = \Sigma_{i=1}^n f_i(v_i)]$

Example: Suppose that there are only two factors: health (H) and wealth (W).

- $u(hw) = 5$
- $u(h\overline{w}) = 2$
- $u(\overline{h}w) = 1$
- $u(\overline{hw}) = 0$

Easy to check: health is utility independent of wealth. There is a multiplicative representation:

- $f_H(h) = 3$; $f_H(\overline{h}) = 1$; $f_W(w) = 2$; $f_W(\overline{w}) = 1$; $c = -1$

There is no additive representation (homework).

Additive Independence

Suppose $Z_1 \cup \ldots \cup Z_k = V$

- $Z_1, \ldots, Z_k$ not necessarily disjoint

$Z_1, \ldots, Z_k$ are *additively independent* (wrt $\preceq_u$) if preferences depend only on the marginal distributions on $Z_i, \ldots, Z_k$, i.e., if

$$p_1^{Z_j} = p_2^{Z_j} \text{ for } j = 1, \ldots, k, \text{ then } p_1 \sim_u p_2$$

Theorem 2: $Z_1, \ldots, Z_k$ are additively independent iff there exist $f_1, \ldots, f_k$ such that

$$u(o) = f_1(o_{Z_1}) + \cdots + f_k(o_{Z_k})$$

H and W are not additively independent. For suppose

- $p_1(hw) = p_1(\overline{h}w) = p_1(h\overline{w}) = p_1(\overline{hw}) = 1/4$
- $p_2(hw) = p_2(\overline{hw}) = 1/2$; $p_2(\overline{h}w) = p_2(h\overline{w}) = 0$

Then $p_1^H = p_2^H$, $p_1^W = p_2^W$, but $p_1 \not\sim p_2$.

- Conclusion: there is no additive representation of u

Conditional Additive Independence

Suppose X, Y, Z is a partition of V.

Definition: X and Y are *conditionally independent* given Z (wrt $\preceq_u$) if, for any fixed $o' \in O_Z$, X and Y are additively independent with respect to $\preceq_u^{o'}$

- Write $CAI(X, Z, Y)$

Example: Imagine X is whether or not you have a computer, Y is whether or not you have Excel, and Z is whether or not you have a particular computer game. It seems reasonable to have Y being utility independent of Z given X. But if Z is whether or not you have Word, Y might not be independent of Z given X

- Word and Excel are arguably more useful together than separately

Theorem 3: $CAI(X, Z, Y)$ iff $X \cup Z$ and $Y \cup Z$ are additively independent.

Corollary: $CAI(X, Z, Y)$ iff there exist f_{XZ} and f_{YZ} such that

$$u(x, y, z) = f_{XY}(x, z) + f_{YZ}(y, z).$$

Graphical Representations of Conditional Additive Independence

Theorem 4: For any utility function u, there is a graph G_u such that $CAI(X, Z, Y)$ iff Z *separates* X from Y in G_u: every path from a node in X to a node in Y passes through some node in Z.

- G_u is called an *independence map* for u

Definition: A *clique* in G is a set of nodes that are completely connected (there is an edge between every pair of them. A *maximal clique* is a clique that is not a subset of any other clique.

Theorem 5: G is an independence map for u iff u has an additive decomposition over the maximal cliques of G. If $Z_1, \ldots, Z_k$ are the maximal cliques of G, then there exist functions $f_1, \ldots, f_k$ such that

$$u(o) = f_1(o_{Z_1}) + \cdots + f_k(o_{Z_k})$$

Advantages of a graphical representation

A graphical representation of utilities has essentially the same advantages as a graphical representation of probability:

- Computational advantages: it's easier to compute expected utility
  - Can combine graphical representations of utility + probability
- advantages for utility elicitation

Now consider a representation even closer to Bayesian networks …

Ceteris Paribus Preferences

Consider a statement like

- I prefer fish to meat, all else being equal

Claim: such *ceteris paribus* statements are how we typically think of preferences

We sometimes want *conditional ceteris paribus* statements:

- I prefer white wine to red, given that we're having fish

Can we represent such qualitative ceteris paribus preferences graphically?

CP Networks

A CP (conditional preference/ceteris paribus) net is a directed graph:

- The nodes are labeled by factors $(V_1, \ldots, V_n)$
- Let $\mathrm{Par}(V_i)$ be the parents of V_i in the graph
- A node V_i is associated with a *conditional preference table* $\mathrm{CPT}(V_i)$
  - $\mathrm{CPT}(V_i)$ has a partial order $\preceq^i_y$ for each element $y \in O_{\mathrm{Par}(V_i)}$.

A CP-net describes a preference order $\succeq$ such that

- if V is a root and $v_i \succeq v_j$ is in the CPT associated with the root, then $(x, v_i, y) \succeq (x, v_j, y)$.
  - All else being equal, $V = v_i$ is preferred to v_j
- If V is an arbitrary node in the tree, if $v_i \succeq v_j$ in the row of the CPT corresponding to a setting x of $\mathrm{Par}(V)$, then $(x, v_i, y) \succeq (x, v_j, y)$.

Example: Suppose there are two variables:

- SOUP, with values fish (f) and vegetable (v)
  - I strictly prefer f to v
- WINE, with values red (r) and white (w)
  - I prefer r to w given v;
  - I prefer w to r given f

Here is the corresponding CP net:

Natural preferences can be expressed this way:

- I prefer a 1000 MHz processor to an 800 MHz processor
- I prefer a 19in screen to a 17in screen if there's a Sony video card
- CPU speed is more important than the CPU manufacturer

CP-nets: Semantics

How do we know that a CP-net describes a preference order?

- It may not be asymmetric

Theorem: Every *acyclic* CP-net whose associated CP tables have strict partial orders (no indifference!) determines a preference order.

Proof: By induction on the number n of variables.

- If $n = 1$: trivial
- In general, consider a root V of the CP-net. Suppose the associated CP table has $v_1 \succ \ldots \succ v_m$.
  - Consider the CP-net N_i obtained by removing V and restricting the CP table of each successor V' of V to $V = v_i$.
  - By induction, we get m preference orders, one corresponding to each N_i, $i = 1, \ldots, m$
  - Combine these preference orders by taking
    - $(x, v_i, y) \succ (x, v_j, y)$ if $v_i \succ v_j$
    - $(x, v_i, y) \succ (x', v_i, y')$ if $(x, y) \succ (x', y')$ in N_i
  - Take transitive closure.
  - Claim: there can't be any cycles.

Acyclicity is necessary to ensure consistency:

Have $a_1b_1 \succ a_1b_2 \succ a_2b_2 \succ a_1b_1$.

- Note that in an acyclic CP-net, variables "higher up" are more important
- Acyclicity $\Rightarrow$ that variables can be ordered in terms of importance.

So is the requirement of no indifference:
Suppose $A \rightarrow B$.

- CPT for A: $a \sim \overline{a}$ CPT for B:
  - given a, $b \succ \overline{b}$
  - given $\overline{a}$, $\overline{b} \succ b$
- Then have $ab \succ a\overline{b} \succeq \overline{ab} \succ \overline{a}b \succeq ab$

However, there are cyclic CP-nets that allow indifference that are consistent.

Consistency checking is NP-complete (hard!) in general

- Many special cases are known to be in polynomial time

Outcome Optimization

Restrict for now to acyclic CP-nets with total orders.

- Determining the best outcome(s) given some instantiation of the variables in $X \subseteq V$ is easy:
  - Can be done in linear time
- Idea: Compute the values of each of the unset variables, starting at the root of the CP net and working down.
  - If $V_i \in X$, do nothing: its value is determined
  - otherwise, compute the best value of V_i given the values of its parents in the best outcome (which are already determined)

Comparing Outcomes

Notation: write $N \models o \succeq o'$ if $o \succeq o'$ in the partial order determined by N.

Theorem: In general, determining if $N \models o \succeq o'$ is NP-hard; if N is a *polytree*, then it can be done in polynomial time.

- N is a polytree if the undirected graph corresponding to N has no cycles

Expressive Power

Not all preference orders are expressible with CP nets.

Example: Suppose you're best off if X and Y have the same value (e.g., you want your clothes to match): Thus,

- $(1,1) \succ (1,0)$; $(1,1) \succ (0,1)$
- $(0,0) \succ (1,0)$; $(0,0) \succ (0,1)$

These ceteris paribus preferences do not tell us how to compare (0,0) and (1,1).

Nevertheless,

- many preferences can be expressed
- people are comfortable thinking in ceteris paribus terms
  - This makes it relatively easy to elicit preferences

Ongoing research:

- enriching the CP net representation to be able to capture more preferences

UCP Nets

- CP-nets make it easy to find optimal outcomes, but hard to test for dominance in general
  - We also can't quantify the tradeoffs between choices
- GAI models make it easy to test for dominance in general, but hard to find optimal outcomes

UCP-nets combine ideas from CP-nets and GAI models.

- Formally, a UCP-net is a CP-net, except the CP table has (conditional) utilities
- A UCP network for $u(X_1, \ldots, X_n)$ consists of
  - a dag G over $X_1, \ldots, X_n$
  - factors $f_i(X_i, \mathrm{Par}(X_i))$ such that

$$u(X_1, \ldots, X_n) = \Sigma_{i=1} f(X_i, \mathrm{Par}(X_i))$$

Example:

Compute utilities assuming GAI:

$$\begin{aligned} u(ab\overline{cd}) &= f_1(a) + f_2(b) + f_3(a, b, \overline{c}) + f_4(\overline{c}, \overline{d}) \\ &= 5 + 5 + .1 + .3 = 10.4 \end{aligned}$$

Advantages of UCP-nets

Not every utility function has a UCP representation, but those that do, have attractive computational properties:

- Can easily compute whether $u(o_1) \geq u(o_2)$
  - Simply sum the values of the factors
- Computing best outcome given constraints is also easy
  - As with CP nets
- Can also combine UCP-net representation of utilities + Bayesian network representation of probabilities to get efficient computation of action with highest expected utility.
- Can often compute approximate solutions particularly efficiently by observing that higher nodes in a UCP-net have a larger impact on utility
  - Can compute when lower nodes can be ignored

UCP nets also have pragmatic advantages when it comes to elicitation.

- eliciting structure is typically easy
- eliciting *normalized* CPTs is relatively easy

- Given arbitrary factors f_X, can replace them by normalized factors g_X:
  - for each value y of $\text{Par}(X)$, $\min_x g_X(x,y) = 0$, $\max_x g_X(x,y) = 1$
- Must still elicit the tradeoffs between factors
  - numbers α_X^y, β_X^y such that $f_X(x,y) = \alpha_X^y g_X(x,y) = \beta_X^y g_X(x,y)$
- Various elicitation techniques have been proposed

A Concrete Example

Should a woman do a prenatal test to check if her child has Down's syndrome?

- Two tests available
  - CVS (chorionic villus sampling)
  - AMNIO (amnioscentesis)
- Both tests carry a risk of causing miscarriage
- The risk is higher with CVS
- CVS is more accurate, can be performed earlier
- Miscarriage (MIS) and elective termination of pregnancy (IAB) reduce chances of future pregnancy

Here is the influence diagram:

Outcomes

The utility can depend on 5 attributes:

- T: type of testing
- D: fetus status
- L: loss of possible pregnancy
- K: knowledge of the fetus's status
- F: future successful pregnancy

There are $3 \times 2 \times 3 \times 3 \times 3 \times 2 = 108$ outcomes.

There may be some independence:

- Maybe $U(T,D,L,K,F) = U(T) + U(L,F) + U(K,L,D)$
- Then can specify U using 27 parameters $(3 + 3 \times 2 + 3 \times 3 \times 2)$ instead of 108

The difficulty of eliciting utility has led to the use of a single "universal" utility function for all patients:

- All women over 35 should get prenatal testing

But this leads to problems:

- Not everyone feels the same about the risk of having a Down's child or having a miscarriage.

The Effect of Framing

[Kahnemann-Tversky, 1981]

Problem 1a: Is surgery or radiation therapy better for lung cancer?
Surgery: Of 100 people having surgery, 90 live through the post-operative period, 68 are alive at the end of the first year, and 34 are alive at the end of five years.

Radiation Therapy: Of 100 people having radiation therapy, all live through the treatment, 77 are alive at the end of the first year, and 22 are alive at the end of five years.

Problem 2a: Assume yourself \$300 richer than you are today. Choose between

- a sure gain of \$100
- 50% chance to gain \$200 and 50% chance to gain nothing

Compound decisions

Problem 3: You must choose one of A and B, and one of C and D. These are simultaneous decisions.

(i) Choose between

A. a sure gain of \$240

B. 25% chance to gain \$1000; 75% chance to gain nothing

(ii) Choose between

C. a sure loss of \$750

D. 75% chance to lose \$1000; 25% chance to lose nothing

Easy to check: B+C dominates A+D:

- A+D: 25% chance to win \$240, 75% chance to lose \$760
- B+C: 25% chance to win \$250, 75% chance to lose \$750

Back to Framing

Problem 1b: Is surgery or radiation therapy better for lung cancer?

Surgery: Of 100 people having surgery, 10 die through the post-operative period, 32 die by the end of the first year, and 66 die by the end of five years.

Radiation Therapy: Of 100 people having radiation therapy, all live through the treatment, 23 die by the end of the first year, and 78 die by the end of five years.

Problem 2b: Assume yourself \$500 richer than you are today. Choose between

- a sure loss of \$100
- 50% chance to lose \$200 and 50% chance to lose nothing

Hidden Dominance

Problem 4a: Which lottery do you prefer?

A:	90% white	6% red	1% green	3% yellow
	\$0	win \$45	win \$30	lose \$15
B:	90% white	7% red	1% green	2% yellow
	\$0	win \$45	lose \$10	lose \$15

Problem 4b: Which lottery do you prefer?

C:	90% white	6% red	1% green	1% blue	2% yellow
	\$0	win \$45	win \$30	lose \$15	lose \$15
D:	90% white	6% red	1% green	1% blue	2% yellow
	\$0	win \$45	win \$45	lose \$10	lose \$15

People aren't always smart enough to be "rational" in the Savage sense.

- It may be too hard computationally for computers too