Redoing the Foundations of Decision Theory

Joe Halpern
Cornell University

Joint work with Larry Blume and David Easley, Economics, Cornell
Savage’s Framework for Decision Theory

Savage assumes that a decision maker (DM) starts with

- a set $S$ of states
- a set $O$ of outcomes
- a preference order $\succeq$ on (Savage) acts – functions from states to outcomes – satisfying certain postulates
  - E.g. transitivity: if $a_1 \succeq a_2$ and $a_2 \succeq a_3$, then $a_1 \succeq a_3$.

Savage proves that if a DM’s preference order satisfies these postulates, then the DM is acting as if

- he has a probability $\Pr$ on states
- he has a utility function $u$ on outcomes
- he is maximizing expected utility:
  - $a \succeq b$ iff $E_{\Pr}[u_a] \geq E_{\Pr}[u_b]$.
  - $u_a(s) = u(a(s))$: the utility of act $a$ in state $s$
Are Savage Acts Reasonable?

Many problems have been pointed out with Savage’s framework. We focus on one:

- How reasonable is it that a DM can completely specify the state space or the outcome space?
  - What are the states/outcomes if we’re trying to decide whether to attack Iraq?
- What are the acts if we can’t specify the state/outcome space?

A related problem: even if we can specify the states/outcomes, there are probably a lot of them.

- How reasonable is it for a DM to have a preference order on $|O|^{|S|}$ acts?
Acts as Programs

Claim: people don’t think of acts as functions:

- We don’t think of the state space and the outcomes when we contemplate the act “Buy 100 shares of IBM”!
- We may think of a procedure:
  - Call the stock broker, place the order, …

An alternative:

- Instead of taking acts to be functions from states to outcomes, acts are syntactic objects
  - essentially, acts are *programs* that the DM can run.
The Setting

Savage assumes that a DM is given a state space and an outcome space. We assume that the DM has

- a set $\mathcal{A}_0$ of primitive programs
  - Buy 100 shares of IBM
  - Attack Iraq

- a set $T_0$ of primitive tests (i.e., formulas)
  - The price/earnings ratio is at least 7
  - The moon is in the seventh house

- a theory $\mathcal{A}_X$
  - Some axioms that describe relations between tests
    - E.g., $t_1 \iff t_2 \land t_3$
Two obvious questions (to a computer scientist!):

- What is the programming language?
- What is the semantics of a program?
The Programming Language

We focus on two programming constructs:

• **if . . . then . . . else**
  
  – If \( a_1 \) and \( a_2 \) are programs, and \( t \) is a test, then
    \[
    \text{if } t \text{ then } a_1 \text{ else } a_2 \text{ is a program}
    \]
  
  – **if** moon in seventh house **then** buy 100 shares IBM

  – Once we allow tests, we need a language in which to express them

• **randomization:**
  
  – If \( a_1 \) and \( a_2 \) are programs and \( r \in [0, 1] \), then
  \[
  r a_1 + (1 - r) a_2 \text{ is a program}
  \]
  
  ○ With probability \( r \) perform \( a_1 \); with probability \( 1 - r \), perform \( a_2 \)

  – People probably don’t use randomized acts
  
  ○ We use them only to compare our results to others in the literature
Programming Language: Syntax

We start with

- a set $A_0$ of primitive acts
  - Buy 100 shares of IBM
  - Attack Iraq
- A set $T_0$ of primitive tests (propositions)
  - The price/earnings ratio is at least 7
  - The moon is in the seventh house

Form more complicated propositions by closing off under conjunction and negation:

- If $t_1$ and $t_2$ are propositions, so are $t_1 \land t_2$ and $\neg t_1$

Form more complicated acts by closing off under if . . . then . . . else and (possibly) randomization.

- Given $A_0$ and $T_0$,
  - let $A$ consist of all acts that can be formed using only the if . . . then . . . else construct;
  - let $A^+$ consist of all acts that can be formed using if . . . then . . . else and randomization
Programming Language: Semantics

Finding appropriate semantics for programming language is a major research topic:

- What should a program mean?

In this paper, we consider input-output semantics:

- A program defines a function from states to outcomes (or probability measures on outcomes if randomization is allowed)
  - a Savage act (Anscombe-Aumann horse lottery)
- The state and outcome spaces are now subjective.
  - Different agents can model them differently
Semantics: Formal Details

Given a state space $S$ and an outcome space $O$, we want to view acts as function from $S$ to $O$. We first need

- a program interpretation $\rho_{SO}$ that associates with each primitive program in $A_0$ a function from $S$ to $O$

We want to extend $\rho_{SO}$ to a function that associates with each program in $A$ a function from $S$ to $O$:

- How do we deal with $\textbf{if } t \textbf{ then } a_1 \textbf{ else } a_2$?
  - if $t$ is true, it’s the function $\rho_{SO}(a_1)$
  - if $t$ is false, it’s the function $\rho_{SO}(a_2)$

But how do we determine if $t$ is true?

We need a test interpretation $\pi_S$ that associates with each primitive proposition in $T_0$ an event (a subset of $S$)

$$\pi_S : T_0 \rightarrow 2^S$$

- Then can extend $\pi_S$ in the obvious way to all tests
  - $t_1 \land t_2$ is true iff both $t_1$ and $t_2$ are true
  - $\neg t$ is true if $t$ isn’t true
Given $S$, $O$, $\rho_{SO}$, $\pi_{S}$, we can extend $\rho_{SO}$ (by the obvious induction) to if . . . then . . . else:

$$\rho_{SO}(\textbf{if} \ t \ \textbf{then} \ a_1 \ \textbf{else} \ a_2)(s) = \begin{cases} 
\rho_{SO}(a_1)(s) & \text{if } s \in \pi_{S}(t) \\
\rho_{SO}(a_2)(s) & \text{if } s \notin \pi_{S}(t)
\end{cases}$$

If we have randomization, then

$$\rho_{SO}^{+} : \mathcal{A}^{+} \rightarrow (S \rightarrow \Delta(O))$$

- $\Delta(O)$ consists of all distributions on $O$
Where We’re Headed

We prove the following type of theorem:

If a DM has a preference order on programs satisfying appropriate postulates, then there exist

- a state space $S$,
- a probability $\Pr$ on $S$,
- an outcome space $O$,
- a utility function $u$ on $O$,
- a program interpretation $\rho_{SO}$,
- a test interpretation $\pi_S$

such that $a \succeq b$ iff $\Pr[E_{\rho_{SO}}(a)] \geq \Pr[E_{\rho_{SO}}(b)]$.

- This is a Savage-like result
  - The postulates are variants of standard postulates
  - The DM has to put a preference order only on “reasonable” acts

But now $S$ and $O$ are subjective, just like $\Pr$ and $u$!
The Benefits of the Approach

We have replaced Savage acts by programs and prove Savage-type theorems. So what have we gained?

• Acts are easier for a DM to contemplate
  – No need to construct a state space/outcome space
  – Just think about what you can do

• Different agents can have completely different conceptions of the world
  – We might agree on the primitive acts but have completely different state spaces
    ○ You might make decision on stock trading based on price/earnings ratio, while I use astrology (and might not even understand the notion of p/e ratio)
    ○ “Agreeing to disagree” results (which assume a common state space) disappear
    ○ (Un)awareness becomes particularly important

• Can deal with unanticipated events, novel concepts:
  – Updating $\neq$ conditioning
To get our “Savage-like” theorem, we have a postulate that guarantees that all programs that act the same as functions are equivalent

- But what if the DM can’t tell that two equivalent programs are equivalent?
  - For rich programming languages, equivalence is undecidable
  - Even for our propositional programming language, it’s co-NP hard (must test equivalence of propositional formulas)
- We do not have to identify programs that act the same as functions

We don’t have to use input-output semantics

- E.g., we can take the semantics of a program to be a sequence of states, followed by an outcome
  - the “path” followed to get to the outcome
- Two programs might have the same input-output semantics, but different “path” semantics
Framing Effects

**Example:** [McNeill et al.] DMs are asked to choose between surgery or radiation therapy as a treatment for lung cancer. They are told that,

- **Version 1:** of 100 people having surgery, 90 alive after operation, 68 alive after 1 year, 34 alive after 5 years; with radiation, all live through the treatment, 77 alive after 1 year, 22 alive after 5 years

- **Version 2:** with surgery, 10 die after operation, 32 dead after one year, 66 dead after 5 years; with radiation, all live through the treatment, 23 dead after one year, 78 dead after 5 years.

Both versions equivalent, but

- In **Version 1**, 18% of DMs prefer radiation;
- in **Version 2**, 44% do
Framing in our Framework

Primitive propositions:

- $RT$: 100 people have radiation therapy;
- $S$: 100 people have surgery;
- $L_0(k)$: $k/100$ people live through operation ($i = 0$)
- $L_1(k)$: $k/100$ are alive after one year
- $L_5(k)$: $k/100$ are alive after five years
- $D_0(k)$, $D_1(k)$, $D_5(k)$ similar, with death

Primitive programs

- $a_S$: perform surgery (primitive program)
- $a_R$: perform radiation therapy
Version 1: Which program does the DM prefer:

\[ a_1 = \text{if } t_1 \text{ then } a_S \text{ else } a, \text{ or } \]
\[ a_2 = \text{if } t_1 \text{ then } a_R \text{ else } a, \]

where \( a \) is an arbitrary program and

\[ t_1 = (S \Rightarrow L_0(90) \land L_1(68) \land L_5(34)) \land \\
(RT \Rightarrow L_0(100) \land L_1(77) \land L_5(22)) \]

Can similarly capture Version 2, with analogous test \( t_2 \) and programs \( b_1 \) and \( b_2 \)

Perfectly consistent to have \( a_1 \succ a_2 \) and \( b_2 \succ b_1 \)

A DM does not have to identify \( t_1 \) and \( t_2 \)

– Preferences should change once \( t_1 \Leftrightarrow t_2 \) is added to theory
The Cancellation Postulate

Back to the Savage framework:

**Cancellation Postulate:** Given two sequences \( \langle a_1, \ldots, a_n \rangle \) and \( \langle b_1, \ldots, b_n \rangle \) of acts, suppose that for each state \( s \in S \)
\[
\{\{a_1(s), \ldots, a_n(s)\}\} = \{\{b_1(s), \ldots, b_n(s)\}\}.
\]
- \( \{\{o, o, o, o', o'\}\} \) is a multiset

If \( a_i \succeq b_i \) for \( i = 1, \ldots, n-1 \), then \( b_n \succeq a_n \).

Cancellation is surprisingly powerful. It implies

- **Reflexivity**

- **Transitivity:**
  
  - Suppose \( a \succeq b \) and \( b \succeq c \). Take \( \langle a_1, a_2, a_3 \rangle = \langle a, b, c \rangle \) and \( \langle b_1, b_2, b_3 \rangle = \langle b, c, a \rangle \).

- **Event independence:**
  
  - Suppose that \( T \subseteq S \) and \( f_Tg \succeq f'_Tg \)
    
    - \( f_Tg \) is the act that agrees with \( f \) on \( T \) and \( g \) on \( S - T \).
  
  - Take \( \langle a_1, a_2 \rangle = \langle f_Tg, f'_Tg' \rangle \) and \( \langle b_1, b_2 \rangle = \langle f'_Tg, f_Tg' \rangle \).
  
  - Conclusion: \( f_Tg' \succeq f'_Tg' \)
Cancellation in Our Framework

An act in our sense (i.e., a program) can be viewed as a function from truth assignments to primitive acts:

- E.g., consider $\text{if } t \text{ then } a_1 \text{ else } (\text{if } t' \text{ then } a_2 \text{ else } a_3)$:
  
  $- t \land t' \rightarrow a_1$
  $- t \land \neg t' \rightarrow a_1$
  $- \neg t \land t' \rightarrow a_2$
  $- \neg t \land \neg t' \rightarrow a_3$

Similarly for every program.

Can rewrite the cancellation postulate using programs:

- replace “outcomes” by “primitive programs”
- replace “states” by “truth assignments”
  
  - i.e., replace $a_i(s)$ by $a_i(v)$, where $v$ is a truth assignment (valuation of primitive tests)
Program Equivalence

When are two programs equivalent?

• That depends on the choice of semantics

• With input-output semantics (i.e., if programs represent functions from states to outcomes), two programs are equivalent if they determine the same functions no matter what $S$, $O$, $\pi_S$, and $\rho_{SO}$ are.

Example 1: (if $t$ then $a$ else $b$) $\equiv$ (if $\neg t$ then $b$ else $a$).

• These programs determine the same functions, no matter how $t$, $a$, and $b$ are interpreted.

Example 2: If $t \equiv t'$, then

$$ (\text{if } t \text{ then } a \text{ else } b) \equiv (\text{if } t' \text{ then } a \text{ else } b). $$

• But testing equivalence of propositional formulas is hard . . .

Lemma: Cancellation $\Rightarrow$ if $a \equiv b$, then $a \sim b$. 
The Main Result

**Theorem:** Given a partial order (reflexive and transitive) \( \succeq \) on acts satisfying Cancellation, there exist

- a set \( S \) of states,
- a set \( \mathcal{P} \) of probability measures on \( S \),
- a set \( O \) of outcomes,
- a utility function \( u \) on \( O \),
- a program interpretation \( \rho_{SO} \),
- a test interpretation \( \pi_S \)

such that

\[
a \succeq b \text{ iff } E_{Pr}[u_a] \geq E_{Pr}[u_b] \text{ for all } Pr \in \mathcal{P}
\]

- \( u_a \) is the random variable such that \( u_a(s) = u(\rho_{SO}(a)(s)) \)

Moreover, if \( \succeq \) is totally ordered, then \( \mathcal{P} \) can be taken to be a singleton.

- We can replace the set of probabilities + utility function with a single probability and a set of utility functions.
Uniqueness

Savage gets uniqueness; we don’t:

- $S$ and $O$ are not unique, but we can find a unique minimal $S^*$ and $O^*$

- In the totally ordered case, $S^*$ can be taken to be a subset of the set of truth assignments.

- Not in the partially ordered case:
  - Even with no primitive propositions, suppose two primitive programs $a$ and $b$ are incomparable.
  - Need two states, two outcomes, and two probability measures to represent this
  - Define
    \[
    a(s_1) = o_1, \quad a(s_2) = o_2 \\
    b(s_1) = o_2, \quad b(s_2) = o_1 \\
    \Pr_1(s_1) = 1 \\
    \Pr_2(s_2) = 1
    \]

- Can’t hope to have a unique probability measure on $S^*$, even in the totally ordered case:
  - there aren’t enough acts to determine it
  - If we just have $a \succ \textbf{if } t \textbf{ then } a \textbf{ else } b \succ b$, then many different probabilities will work
Adding Randomization

If we allow randomization in programs, Cancellation gives us independence for rational coefficients:

**Lemma**: Cancellation implies $f \succeq g$ iff for all rational $\alpha \in [0, 1]$ and all $h$, $\alpha f + (1 - \alpha)h \succeq \alpha g + (1 - \alpha)h$.

Get independence for all coefficients by assuming an appropriate Archimedean axiom:

(a) If $f \succ g \succ h$ then there exist $0 < \alpha, \beta < 1$ such that

$$f \succ \alpha f + (1 - \alpha)h \succ g \succ \beta f + (1 - \beta)h \succ h.$$ 

(b) $\{\alpha \in [0, 1] : \alpha f + (1 - \alpha)g \succeq \alpha f + (1 - \alpha)h\}$ is closed.

Get representation theorem for $\mathcal{A}^+$ assuming Cancellation and the Archimedean postulate. Moreover, if the order is total, then the expected utility of acts is unique up to affine transformations.

- For any two representations, the expected utility of acts agree up to an affine transformation
Fixing the Outcome Space

In some applications, it makes sense to assume a fixed, “objective” outcome space $O$.

- e.g., in financial applications, the outcome space can be $\mathbb{R}

In this case, it seems reasonable to assume that, among the primitive acts, there are constant acts:

- For each $o \in O$, there is an act $\sigma$
  
  - Semantically, given $S$, $\sigma$ will be interpreted as the constant function on $S$ that always returns $o$

We need (again, standard) postulates to guarantee that constant acts are really constant. E.g.:

$$\sigma_1 \succeq \sigma_2 \text{ implies if } t \text{ then } \sigma_1 \text{ else } a \succeq \text{ if } t \text{ then } \sigma_2 \text{ else } a$$

Can again prove a representation theorem, if the language allows randomization. Moreover, we get uniqueness of the probability measure.

- Getting a representation theorem with a fixed outcome space and no randomization remains an open problem
Updating

In the representation, can always take the state space to have the form $AT_{AX} \times TOT(\succeq)$:

- $AT_{AX} =$ all truth assignments to tests compatible with the axioms $AX$
- $TOT(\succeq) =$ total orders extending $\succeq$

Updating proceeds by conditioning:

- Learn $t \Rightarrow$ representation is $\mathcal{P} \mid t$
- Learn $a \succeq b$: representation is $\mathcal{P} \mid (\succeq \oplus (a, b))$
Non-classical DMs

We have assumed that DMs obey all the axioms of propositional logic

\[ \pi_S(\neg t) = S - \pi_S(t) \quad \text{and} \quad \pi_S(t_1 \land t_2) = \pi_S(t_1) \cap \pi_S(t_2). \]

But we don’t have to assume this!

• Instead, write down explicitly what propositional properties hold

• We still get that Cancellation, and that \( a \equiv b \) implies \( a \sim b \)

• But now this isn’t so bad: intuitively, the logic is restricted so that if \( a \equiv b \), then the DM can tell that \( a \) and \( b \) are equivalent, and so we should have \( a \sim b \)
Conclusions

The theorems we have proved show only that this approach generalizes the classic Savage approach.

- The really interesting steps are now to use the approach to deal with issues that the classical approach can’t deal with
  - conditioning on unanticipated events
  - (un)awareness
  - …