Decision-Making & Modelling Under Uncertainty (DMU)

Oliver Maclaren (oliver.maclaren@auckland.ac.nz)

[10 lectures/tutorials]

- Decision-making under uncertainty [5/10]
  - Basic concepts
  - Risk, probability, utility
  - Statistical: extended setup
    - formulation & empirical risk approx.
    - minimax & Bayes
  - Tutorial sheet
- Modelling under uncertainty [5/10] { models of risk & intervention
  - probability, graphical models, & independence
  - causal interpretations of graphical models
  - stochastic process models (esp. Markov)
  - simulation & estimation tools
  - Tutorial sheet

Lecture 6: DAGs cont'd

- d-separation
- interventions & causality
- defining target queries for probabilistic & causal models

Appendices:

- Computing answers to queries
- 'Causal' vs 'evidential' decision theory

[History of DAGs

- Wright } 1930s
- Pearl
- Glymour, Scheines, Spirtes } 1980s on

etc]

Recall:

Given these definitions, a DAG directly encodes conditional independences of the form:

$$X \perp \text{Pred}(X) \setminus \text{Pa}(X) \mid \text{Pa}(X)$$

where $\text{Pred}(X)$ = 'Predecessors of X'
$\text{Pa}(X)$ = 'Parents of X'
$\text{Pred}(X) \setminus \text{Pa}(X)$ = 'non parent predecessors'

ie 'X is independent of its non-parent predecessors, given its parents'

(only the immediate predecessors, ie parents, matter)

$$\Rightarrow P(x \mid \text{pred}(x)) = P(x \mid \text{pa}(x))$$

[where pa(x) stands for 'values of parents of X' etc.]

Recall:

Other independencies in DAGs

A given DAG tells us 'directly' about the (probabilistic) dependencies of a node on its ancestors ('past' nodes)

→ it doesn't 'directly' tell us about the (probabilistic) dependencies of a node on its descendents!

eg $X \to Y$

tells us Y depends on X

& $P(x,y) = P(y|x)P(x)$

($P(y|x)$: $\text{Pa}(y)$; $P(x)$: no parents)

But $P(x,y) = P(x|y)P(y)$ always] prob. theory

& $P(x|y) = \frac{P(y|x)P(x)}{P(y)} \neq P(x)$

$\Rightarrow$ X is not indep. of Y in $X \to Y$

Recall

Causal vs probabilistic dependence?

- The DAG interpretation so far is purely probabilistic, not 'causal'

- We can add interpretational components so that $X \to Y$ also captures 'X causes Y'...

  BUT for now just focus on the probabilistic conditional (in)dependencies implied by a DAG

→ combine ordered Markov factorisation & probability theory

--- OR use graph theory!

Further conditional independences implied by a DAG: directed separation

To determine the other independences implied by the directed Markov factorisation (encoded by the DAG) & the rules of probability, we can consider the graphical concept of

'd-separation'

(for directed separation. Pearl: 1980s)

→ For this we will call any sequence of edges between two nodes, regardless of direction, a dependence path or just 'path':

$A \to B \leftarrow C$ } $A \to B, B \leftarrow C$ is a path

d-separation : probabilistic (in)dependencies implied by DAGs

[More complex than u-separation unfortunately !]

First, a **Definition**:

path blocking

A (dependence) path between any two nodes X, Y in a DAG is 'blocked' by a set of nodes $\mathcal{B}$ iff the path contains at least one sub-path ('junction') of the form:

1. $A \to B \to C$ (chain) where $B \in \mathcal{B}$

or 2. $A \leftarrow B \to C$ (fork) where $B \in \mathcal{B}$

or 3. $A \to D \leftarrow C$ (collider) where $D \notin \mathcal{B}$ & $des(D) \notin \mathcal{B}$

↑ descendents of D

d-separation:

If a set of nodes $\mathcal{B}$ blocks every path between two nodes X & Y then X & Y are said to be 'd-separated given $\mathcal{B}$'

→ According to the standard 'directed Markov' interpretation of a DAG & standard probability theory, we have:

$X \perp\!\!\!\perp Y \mid \mathcal{B}$

$\Leftrightarrow$

X & Y d-separated given $\mathcal{B}$

→ Furthermore:

d-separation can derive all the conditional independences implied by the DAG!

Examples

1. $X \to U \to V \to W \to Y$

$\{U\}, \{U,V\}, \{U,W\}, \{U,V,W\}$

$\{V,W\}$ etc

are all sets that block

$X \to \cdots Y$

ie $X \perp\!\!\!\perp Y \mid \{U\}$ etc.

2. $X \leftarrow U \to Y$

$\{U\}$ d-separates X & Y

$\Rightarrow X \perp\!\!\!\perp Y \mid \{U\}$ (or $X \perp\!\!\!\perp Y \mid U$)

3. $X \to U \leftarrow Y$

U does not separate X&Y

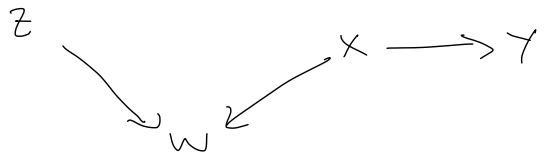
$X \perp\!\!\!\perp Y$ already

but $X \not\perp\!\!\!\perp Y \mid U$ } 'conditioning on a collider' → bad!

[colliders automatically block dependence]

Examples

Consider:



- what can we say about the (in)dependence of Z & Y?

→ we can 'condition' on the empty set to get 'unconditional' independence properties

→ note that the dependence path between Z & Y contains the sub path $Z \to W \leftarrow X$, ie a 'collider'

→ Hence $Z \perp\!\!\!\perp Y \mid \{\} \Leftrightarrow Z \perp\!\!\!\perp Y$

But $Z \not\perp\!\!\!\perp Y \mid W$ } 'unblocks' path

Note: If two nodes are not d-separated, they are called d-connected & are (likely) 'not independent' ie '$X \not\perp\!\!\!\perp Y$'

d-separation for sets of nodes

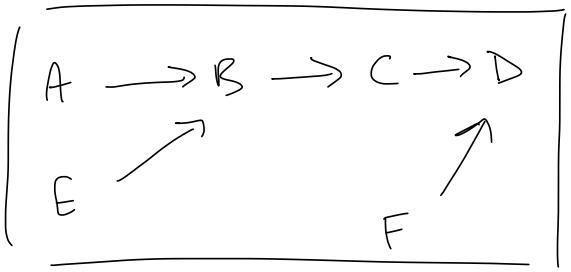
" If A, B & C are (disjoint) sets of nodes (vertices) then A & B are d-separated given C if for every node $X \in A$ & node $Y \in B$, X & Y are d-separated given C "

---

ie all nodes in A are d-separated from all nodes in B given C

→

Eg recall:



Encodes independencies such as:

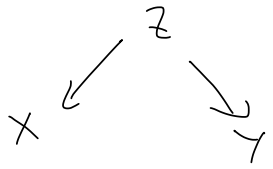
$$C \perp \{A, E\} \mid B$$

ie

$$P(c \mid a, b, e) = P(c \mid b)$$

$\Leftrightarrow$ $\{A, E\}$ & $\{C\}$ are d-separated given $\{B\}$

Exercises:

1. Consider 

which of these is true:

$X \perp\!\!\!\perp Y$

$X \perp\!\!\!\perp Y \mid Z$

2. Suppose in (1) we are sampling from a population & measuring variables:

Z = 'age'

X = 'retirement savings'

Y = 'number of wrinkles'

What can you say about the relationship between retirement savings & wrinkles?

How should you analyse this relationship to understand the 'causal effects'?

3. Consider $A \to B \to C$, $A \leftarrow B \to C$ & $A \leftarrow B \leftarrow C$

- Derive the implied conditional independencies using d-separation
- what do you notice?

4. Consider: 

Is $A \perp\!\!\!\perp C$?

Is $A \perp\!\!\!\perp C \mid B$?

5. In (4), suppose:

- A represents 'height'
- B represents 'basketball success'
- C represents 'basketball shooting skill'

Interpret the associated conditional independencies.

Interventions & causality

Recall $X \to Y$ & $X \leftarrow Y$

encode the same conditional independencies!

similarly $X \to Y \to Z$ (1)

& $X \leftarrow Y \to Z$ (2)

→ would like to distinguish in order to interpret 'causally'

eg X causes Y causes Z (1)

$\neq$

Y causes both X & Z. (2)

Interventions : 'doing' vs 'seeing' ('viewing')

Pearl and others:

$p(y|x)$ = "prob. of $Y=y$ given that we see $X=x$"

→ not causal, might be just a correlation / association etc

want:

$p(y|do(x))$ = "prob. of $Y=y$ given that we do $X=x$"

→ ie we intervene on the system and cause X to have a value, then look at effect on Y

Intervention in a DAG

- $do(X)$: remove all incoming arrows into X (remove all other causes so we can set)
- $do(x) = do(X=x)$: remove arrows & set $X = x$

Example:

$X \to Y$ $\}$ $do(X)$: $X \to Y$

$X \leftarrow Y$ $\}$ $do(X)$: $X \quad Y$

different!

→ adding 'intervention' ('$do(X)$') concept to DAGs distinguishes $X \to Y$ & $X \leftarrow Y$!

Causality defined? (Pearl):

Given a DAG G & associated joint distribution, $P_G(x, y, \ldots)$ ['P_G' = 'P factorises according to G']

Define the causal distribution of Y given X as:

(A) $P_G(y \mid do(x)) = P_{G_X}(y \mid x)$ causal distribution

where G_X is the graph obtained by deleting all arrows pointing into X & $P_{G_X}(x, y, \ldots)$ is the associated probability distribution for G_X, related to $P_G(x, y, \ldots)$ by:

(B) $P_{G_X}(z \mid pa_G(z)) = P_G(z \mid pa_{G_X}(z))$ invariance condition

for all variables z (incl. X & Y) & $pa_{G_X}(z)$ are the parents of z in G_X

Note: $P(z \mid pa(z)) \equiv P(z)$ if $pa(z)$ empty, ie no parents

causal effect = 'do then see'

↑ intervene $G \to G_x$

↑ compute probabilities under G_x, assuming invariance condition to relate to G

Called the 'causal Markov' interpretation of DAG

↳ combines probability & causation

Examples

A) $G: X \to Y$

$\Rightarrow G_x : X \to Y$

$\Rightarrow P_G(y|do(x)) = P_{G_x}(y|x)$

$= P_{G_X}(y|pa_G(y))$

$= P_G(y|pa_{G_X}(y))$

$= P_G(y|x)$

B) $G: X \leftarrow Y$

$\Rightarrow G_x: X \quad Y$ (no arrows)

$\Rightarrow P_G(y|do(x)) = P_{G_x}(y|x)$

$= P_{G_X}(y)$ [$y \perp\!\!\!\perp x$ in G_x]

$= P_{G_X}(y|pa_G(y))$

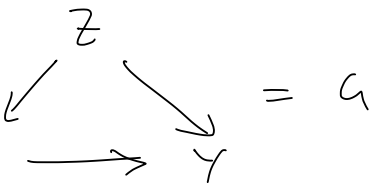
$= P_G(y|pa_{G_X}(y))$

$= P_G(y)$

(y has no parents in G or G_x)

($P(y|Pa(y)) = P(y)$ if $Pa(y) = \phi$)

Exercises

6. Consider: 

$= G$

& $P = P_G$ the associated prob. model

Show:

- $P(y|x) = \int P(y|x,z)P(z|x)dz$

- $P(y|do(x)) = \int P(y|x,z)P(z)dz$

$\neq P(y|x)$.

Note

The expression for $P(y|do(x))$ above is called the 'adjustment formula'

→ computes distribution capturing 'effect of X on Y, accounting for the confounder Z'

Answer to 6

- $P(y|x) = \frac{P(y,x)}{P(x)}$
  - $P(y,x) = \int P(y,x,z)\,dz$

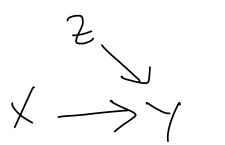
$$= \int P(y|x,z)P(z|x)P(x)\,dz$$

$$= \int P(y|x,z)P(z|x)\,dz \cdot \underbrace{P(x)}_{\text{indep of } z.}$$

$$\Rightarrow \frac{P(y,x)}{P(x)} = P(y|x) = \int P(y|x,z)P(z|x)\,dz \;\checkmark$$

- $P(y|do(x)) = P_{G_x}(y|x)$

& G_x =



from prob. theory def. eg above

$$\Rightarrow P_{G_x}(y|x) = \int P_{G_x}(y|x,z)P_{G_x}(z|x)\,dz$$

- $P_{G_x}(z|x) = P_{G_x}(z \mid pa_{G_x}(z)) = P_G(z \mid pa_{G_x}(z)) = P_G(z)$
- $P_{G_x}(y|x,z) = P_{G_x}(y|pa_G(y)) = P_G(y \mid pa_{G_x}(y)) = P_G(y|x,z)$

$$\Rightarrow \underbrace{P_{G_x}(y|x)}_{P(y|do(x))} = \int P(y|x,z)P(z)\,dz \;\checkmark$$

$(P_G = P)$

Automated answers to 'queries' of a DAG?

Given joint distribution $P(x_1, x_2, \ldots x_n)$ over RVs $X_1, X_2, \ldots, X_n$ implied by a DAG G, we can now answer probabilistic & causal questions by computing:

- $P(\text{Interest variables} \mid \text{observed vars})$ 「probabilistic dependence」
- $P(\text{interest vars} \mid do(\text{observed vars}))$ 「causal dependence」

where

- other variables are marginalised out:

$$P(x,y) = \int P(x,y,z)\,dz$$

- $do(x)$: delete arrows into x etc
- $P(y|x) = \frac{P(y,x)}{P(x)} = \frac{P(y,x)}{\int P(y,x)\,dy}$

Appendix A: computational implementation

Exact inference (computation of answers to queries) is 'NP-hard'
→ use approximate answers!

Simple, naive approach (need better for high dimensions): direct sampling from ordered conditionals $P(x_i | pa(x_i))$

(Naive Monte Carlo)

(see attached →)

Briefly: Given any sorted list $X_1, \dots, X_n$ where all parents of X_i appear before X_i in list: → see over

For $i = 1$ to N

Sample x_i (realisation) value from $P(X_i | pa(x_i))$ (1D dist given parent values; X_i: random; $pa(x_i)$: values of parents)

End

→ samples joint via series of simpler dist. (assume can sample these)

Querying joint

Given sample i from joint eg $(x_1^i, x_2^i, x_3^i \dots x_n^i)$

↳ marginalise vars by 'ignoring' their values:

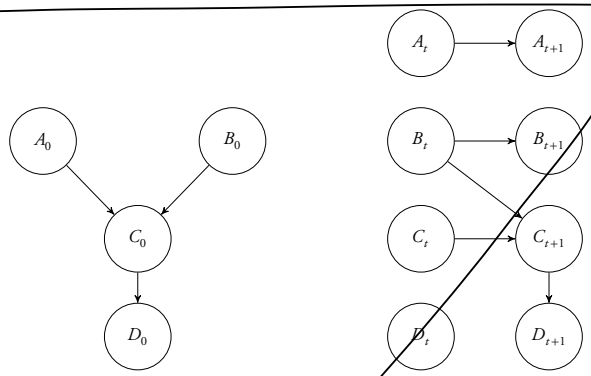
$(x_1^i, x_2^i, \dots x_n^i) \rightarrow (x_1^i, x_3^i)$ sample from $P(x_1, x_3)$

↳ condition on vars by 'restricting attention' to samples with required values

$(x_1^i, x_2^i, \dots x_n^i) \rightarrow$ ignore if $x_2^i \neq 3$ (conditioning on $x_2^i = 3$)

↳ calculate probabilities by counting proportion of cases within marginalised & conditioned samples

→ $\frac{\#\, x_4 = 1 \wedge x_2 = 3}{\#\, x_2 = 3}$ proportion among $x_2 = 3$ that satisfy $x_4 = 1$, ignoring rest of values.



(a) Initial distribution. (b) Transition distribution.

Figure 2.8 Dynamic Bayesian network.

noisy measurements of the altitude; the vertical rate cannot be observed directly. The observation at time t is modeled as coming from a linear Gaussian distribution:

$$p(\mathbf{o}_t \mid \mathbf{s}_t) = \mathcal{N}\left(\mathbf{o}_t \;\middle|\; \begin{bmatrix}1 & 0\end{bmatrix}\mathbf{s}_t, \Sigma\right). \tag{2.21}$$

The covariance matrix Σ, in this case a single-element matrix, controls the measurement noise.

Stationary temporal models involving multiple state variables can be compactly represented using a *dynamic Bayesian network*. A dynamic Bayesian network is composed of two Bayesian networks, one representing the initial distribution and the other representing the transition distribution. The transition distribution is represented by a Bayesian network with two slices. The first slice represents the variables at time t, and the second slice represents the variables at time $t+1$. Figure 2.8 shows an example dynamic Bayesian network with four state variables.

2.2 Inference

The previous section explained how to represent probability distributions. We now discuss how to use these probabilistic representations to perform *inference*. Inference involves determining the distribution over one or more unobserved variables given the values associated with a set of observed variables. For example, suppose we want to

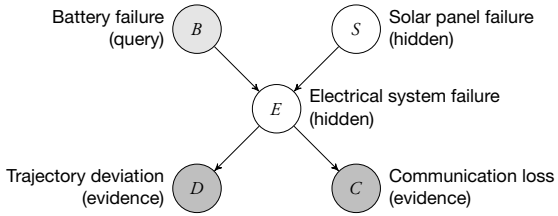


Figure 2.9 Query, evidence, and hidden variables in a Bayesian network.

infer the distribution $P(B \mid d^1, c^1)$ using the satellite Bayesian network introduced in Section 2.1.4. In this case, B is the *query* variable, D and C are the *evidence* variables, and S and E are the *hidden* variables. After a discussion of a couple examples in which inference can be helpful, this section explains how to leverage the structure inherent in a Bayesian network to make efficient inferences.

2.2.1 Inference for Classification

Inference can be used for *classification* tasks, where we want to infer the class given a set of observations or features. For example, suppose we want to determine whether a radar target is either a bird or an aircraft given properties of the radar track. In this case, the class is either bird or aircraft, and the observations might include measurements of the velocity and the amount of fluctuation in the heading over the duration of the track. Most aircraft travel faster than most birds, but there is some overlap, especially with smaller, lower performance aircraft. Migrating birds tend to maintain their heading, in contrast to maneuvering aircraft.

A simple probabilistic model often used in classification tasks is the *naive Bayes* model, which has the structure shown in Figure 2.10. An equivalent but more compact representation is shown in Figure 2.11 using a *plate*, shown as a rounded box. The $i = 1:n$ in the bottom of the box specifies that the i in the subscript of the variable name is repeated from 1 to n.

In the naive Bayes model, the class C is the query variable, and the observed features $O_1, \ldots, O_n$ are the evidence variables. For compactness throughout this book, we will use colon notation occasionally in subscripts. For example, $O_{1:n}$ is a compact way to write $O_1, \ldots, O_n$. The naive Bayes model is called naive because it assumes conditional independence between the evidence variables given the class. Using the notation introduced in Section 2.1.5, we can say $(O_i \perp O_j \mid C)$ for all $i \neq j$. Of course, if these conditional independence assumptions do not hold, then we can add the necessary directed edges between the observed features.

2.2.5 Approximate Inference

One of the simplest approaches to approximate inference involves sampling from the joint distribution represented by the Bayesian network. The first step involves finding a *topological sort* of the nodes in the Bayesian network. A topological sort of nodes in a directed acyclic graph is an ordered list such that if there is an edge $A \rightarrow B$, then A comes before B in the list. For example, a topological sort for the network in Figure 2.9 is B, S, E, D, C. A topological sort always exists, but it may not be unique. Another topological sort for the network in Figure 2.9 is S, B, E, C, D. Algorithm 2.3 provides an algorithm for finding a topological sort of a graph G.

Algorithm 2.3 Topological sort

```
1: function TopologicalSort(G)
2:     n ← number of nodes in G
3:     L ← empty list
4:     for i ← 1 to n
5:         X ← any node not in L but all of whose parents are in L
6:         Add X to end of L
7:     return L
```

Once we have a topological sort, we can begin sampling from the conditional probability distributions. Suppose our topological sort results in the ordering $X_{1:n}$. Algorithm 2.4 shows how to sample from a Bayesian network B. In Line 4, we draw a sample from the conditional distribution associated with X_i given the values of the parents that have already been assigned. Because $X_{1:n}$ is a topological sort, we know that all the parents of X_i have already been instantiated, allowing this sampling to be done.

Algorithm 2.4 Direct sampling from a Bayesian network

```
1: function DirectSample(B)
2:     X_{1:n} ← a topological sort of nodes in B
3:     for i ← 1 to n
4:         x_i ← a random sample from P(X_i | pa_{x_i})
5:     return x_{1:n}
```

'Monte Carlo' (direct)

Table 2.4 shows ten random samples from the network in Figure 2.9. We are interested in inferring $P(b^1 \mid d^1, c^1)$. Only two of the ten samples (pointed to in the table) are consistent with the observations d^1 and c^1. One sample has $B = 1$ and the other sample has $B = 0$. From these samples, we infer that $P(b^1 \mid d^1, c^1) = 0.5$. Of course, we would want to use more than just two samples to accurately estimate $P(b^1 \mid d^1, c^1)$.

→ condition: restrict attention to] !
marginalise: ignore

→ $P(b^1 \mid d^1, c^1) \approx \sum_i \left(\mathbb{1}_{b^i=1} \cdot \mathbb{1}_{d^i=1} \cdot \mathbb{1}_{c^i=1} \right)$

$\sum_i \left(\mathbb{1}_{d^i=1} \cdot \mathbb{1}_{c^i=1} \right)$

Table 2.4 Direct samples from a Bayesian network.

B	S	E	D	C	
0	0	1	1	0	
0	0	0	0	0	
1	0	1	0	0	
1	0	1	1	1	←
0	0	0	0	0	
0	0	0	1	0	
0	0	0	0	1	
0	1	1	1	1	←
0	0	0	0	0	
0	0	0	1	0	

of those (conditional on) that have $c=1, d=1$, ½ have $b=1$

where $\mathbb{1}_{b^i=1} = \begin{cases} 1 \text{ if } b^i=1 \\ 0 \text{ if } b^i \neq 1 \end{cases}$ etc.

ie counting

The problem with direct sampling is that we may waste a lot of time generating samples that are inconsistent with the observations, especially if the observations are unlikely. An alternative approach is called *likelihood weighting*, which involves generating weighted samples that are consistent with the observations. We begin with a topological sort and sample from the conditional distributions in sequence. The only difference in likelihood weighting is how we handle observed variables. Instead of sampling their values from a conditional distribution, we assign variables to their observed values and adjust the weight of the sample appropriately. The weight of a sample is simply the product of the conditional probabilities at the observed nodes. Algorithm 2.5 summarizes this process for a Bayesian network B and observations $o_{1:n}$. If o_i is not observed, then $o_i \leftarrow$ nil.

Algorithm 2.5 Likelihood-weighted sampling from a Bayesian network

```
1: function LikelihoodWeightedSample(B, o_{1:n})
2:     X_{1:n} ← a topological sort of nodes in B
3:     w ← 1
4:     for i ← 1 to n
5:         if o_i = nil
6:             x_i ← a random sample from P(X_i | pa_{x_i})
7:         else
8:             x_i ← o_i
9:             w ← w × P(x_i | pa_{x_i})
10:    return (x_{1:n}, w)
```

Table 2.5 Likelihood weighted samples from a Bayesian network.

B	S	E	D	C	Weight
1	0	1	1	1	$P(d^1 \mid e^1)P(c^1 \mid e^1)$
0	1	1	1	1	$P(d^1 \mid e^1)P(c^1 \mid e^1)$
0	0	0	1	1	$P(d^1 \mid e^0)P(c^1 \mid e^0)$
0	0	0	1	1	$P(d^1 \mid e^0)P(c^1 \mid e^0)$
0	0	1	1	1	$P(d^1 \mid e^1)P(c^1 \mid e^1)$

Table 2.5 shows five likelihood-weighted samples from the network in Figure 2.9. We sample from $P(B)$, $P(S)$, and $P(E \mid B, S)$, as we would with direct sampling. When we come to D and C, we assign $D = 1$ and $C = 1$. If the sample has $E = 1$, then the weight is $P(d^1 \mid e^1)P(c^1 \mid e^1)$; otherwise, the weight is $P(d^1 \mid e^0)P(c^1 \mid e^0)$. If we assume

$$P(d^1 \mid e^1)P(c^1 \mid e^1) = 0.95 \tag{2.44}$$

$$P(d^1 \mid e^0)P(c^1 \mid e^0) = 0.01 \tag{2.45}$$

then we may approximate from the samples in Table 2.5

$$P(b^1 \mid d^1, c^1) \approx \frac{0.95}{0.95 + 0.95 + 0.01 + 0.01 + 0.95} \tag{2.46}$$

$$= 0.331. \tag{2.47}$$

Although likelihood weighting makes it so that all samples are consistent with the observations, it can still be wasteful. Consider the simple chemical detection Bayesian network shown in Figure 2.15, and assume that we detected a chemical of interest. We want to infer $P(c^1 \mid d^1)$. Because this network is small, we can easily compute this probability exactly by using Bayes' rule:

$$P(c^1 \mid d^1) = \frac{P(d^1 \mid c^1)P(c^1)}{P(d^1 \mid c^1)P(c^1) + P(d^1 \mid c^0)P(c^0)} \tag{2.48}$$

$$= \frac{0.999 \times 0.001}{0.999 \times 0.001 + 0.001 \times 0.999} \tag{2.49}$$

$$= 0.5. \tag{2.50}$$

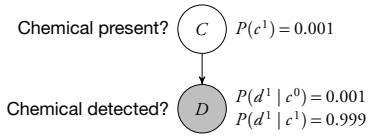


Figure 2.15 Chemical detection Bayesian network.

If we use likelihood weighting, then 99.9% of the samples will have $C = 0$ with a weight of 0.001. Until we get a sample of $C = 1$, which has an associated weight of 0.999, our estimate of $P(c^1 \mid d^1)$ will be 0.

An alternative approach is to use *Gibbs sampling*, which is a kind of *Markov chain Monte Carlo* technique. Unlike the other sampling methods discussed so far, the samples produced by this method are not independent. The next sample depends probabilistically on the current sample, and so the sequence of samples forms a Markov chain. It can be proven that, in the limit, samples are drawn exactly from the joint distribution over the unobserved variables given the observations.

The initial sample can be generated randomly with the observed variables set to their observed values. Algorithm 2.6 outlines how to generate a new sample $x'_{1:n}$ from an existing sample $x_{1:n}$, given a Bayesian network B and observations $o_{1:n}$. Unlike direct sampling, we can use any ordering for the nodes in the network; the ordering need not be a topological sort. Given this ordering, update the sample one variable at a time given the values of the other variables. To generate the value for x'_i, we sample from $P(X_i \mid x'_{1:n\setminus i})$, where $x'_{1:n\setminus i}$ represents the values of all the other variables except X_i. To compute the distribution $P(X_i \mid x'_{1:n\setminus i})$ for a Bayesian network B, we can use Algorithm 2.7. The computation can be done efficiently because we only need to consider the Markov blanket of variable X_i (Section 2.1.5).

Algorithm 2.6 Gibbs sampling from a Bayesian network

```
function GibbsSample(B, o_{1:n}, x_{1:n})
    X_{1:n} ← an ordering of nodes in B
    x'_{1:n} ← x_{1:n}
    for i ← 1 to n
        if o_i = nil
            x'_i ← a random sample from P(X_i | x'_{1:n\i})
        else
            x'_i ← o_i
    return x'_{1:n}
```

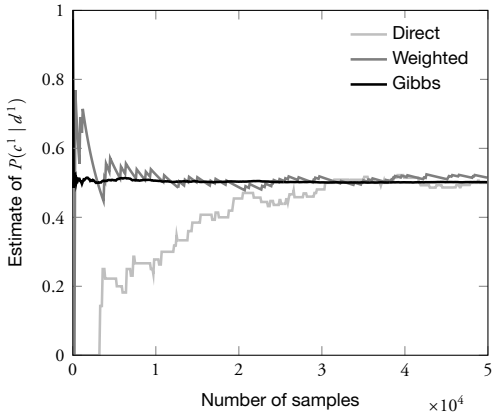


Figure 2.16 Bayesian network sampling methods.

Algorithm 2.7 Distribution at a node given observations at all other nodes

```
function DistributionAtNode(B, X_i, x_{1:n\i})
    𝒯 ← all conditional probability tables associated with B involving X_i
    Remove rows that are inconsistent with x_{1:n\i} from all the tables in 𝒯
    T ← product of the tables remaining in 𝒯
    P(X_i | x_{1:n\i}) ← normalize T
    return P(X_i | x_{1:n\i})
```

Figure 2.16 compares the convergence of the estimate of $P(c^1 \mid d^1)$ using direct, likelihood weighted, and Gibbs sampling. Direct sampling takes the longest to converge. The direct sampling curve has long periods during which the estimate does not change because samples are inconsistent with the observations. Likelihood-weighted sampling converges faster in this example. Spikes occur when a sample is generated with $C = 1$ and then gradually decrease. Gibbs sampling, in this example, quickly converges to the true value of 0.5.

As mentioned earlier, Gibbs sampling, like other Markov chain Monte Carlo methods, produces samples from the desired distribution *in the limit*. In practice, we have to run Gibbs for some amount of time, called the *burn-in period*, before converging to a steady state distribution. The samples produced during burn-in are normally discarded. In addition, because of potential correlation between samples, it is common to *thin* the samples by only keeping every kth sample.

Other approximate inference methods do not involve generating samples. For example, a form of belief propagation called *loopy belief propagation* can be used in networks with undirected cycles for approximate inference. Although not guaranteed to be exact, loopy belief propagation tends to work well in practice and is becoming one of the most popular methods for approximate inference in Bayesian networks.

2.3 Parameter Learning

So far in this chapter, we have assumed that the parameters and structure of our probabilistic models were known. This section addresses the problem of learning the parameters of the model from data.

2.3.1 Maximum Likelihood Parameter Learning

Suppose the random variable C represents whether a flight will result in a mid-air collision, and we are interested in estimating the distribution $P(C)$. Because C is either 0 or 1, it is sufficient to estimate the parameter $\theta = P(c^1)$. What we want to do is infer θ from data D. Let us say that we have a historical database spanning a decade, and let us say we know there were n flights and m mid-air collisions. Our intuition, of course, tells us that a good estimate for θ given the data D is m/n. This estimate corresponds to the *maximum likelihood estimate*,

$$\hat{\theta} = \arg\max_{\theta} P(D \mid \theta) \tag{2.51}$$

The probability of m mid-air collisions out of n flights is given by the *binomial distribution*:

$$P(D \mid \theta) = \frac{n!}{m!(n-m)!}\theta^m(1-\theta)^{n-m} \tag{2.52}$$

$$\propto \theta^m(1-\theta)^{n-m}. \tag{2.53}$$

The maximum likelihood estimate $\hat{\theta}$ is the value for θ that maximizes Equation (2.53). Maximizing Equation (2.53) is equivalent to maximizing the logarithm of the likelihood, often referred to as the *log-likelihood* and often denoted $\ell(\theta)$:

$$\ell(\theta) \propto \ln\left(\theta^m(1-\theta)^{n-m}\right) \tag{2.54}$$

$$= m\ln\theta + (n-m)\ln(1-\theta). \tag{2.55}$$

We can use the standard technique for finding the maximum of a function by setting the first derivative of ℓ to 0 and then solving for θ. The derivative is given by

$$\frac{\partial\ell(\theta)}{\partial\theta} = \frac{m}{\theta} - \frac{n-m}{1-\theta}. \tag{2.56}$$

Appendix B

Causal decision theory? Two kinds of expected utility?

$$\mathbb{E}_{d_i}[u(d_i, s)] = \begin{cases} \int u(d_i, s)\, P(s \mid d_i)\, ds & (1) \\ \int u(d_i, s)\, P(s \mid do(d_i))\, ds & (2) \end{cases}$$

(1): 'evidential decision theory' (Jeffrey, 1965)

(2): 'causal decision theory' (Gibbard & Harper, 1976)

Don't always agree!

→ see attached.

Titelbaum (2022) 'Fundamentals of Bayesian Epistemology v2'

standard decision theory, the example is now known as Allais' Paradox. Allais thought the example revealed a deep flaw in the decision theories we've been considering.

We have been discussing these decision theories as *normative* accounts of how *rational* agents behave. Economists, however, often assume that decision theory provides an accurate *descriptive* account of *real* agents' market decisions. Real-life subjects' responses to cases like the Allais Paradox prompted economists to develop new descriptive theories of agents' behavior, such as Kahneman and Tversky's Prospect Theory (Kahneman and Tversky 1979; Tversky and Kahneman 1992). More recently, Buchak (2013) has proposed a generalization of standard decision theory that accounts for risk aversion without positing declining marginal utilities and is consistent with the Allais preferences subjects often display.

7.3 Causal Decision Theory

Although we have been focusing on the expected values of propositions describing acts, Jeffrey's valuation function can be applied to any sort of proposition. For example, suppose my favorite player has been out of commission for weeks with an injury, and I am waiting to hear whether he will play in tonight's game. I start wondering whether I would prefer that he play tonight on not. Usually it would make me happy to see him on the field, but there's the possibility that he will play despite his injury's not being fully healed. That would definitely be a bad outcome. So now I combine my credences about states of the world (is he fully healed? is he not?) with my utilities for the various possible outcomes (plays fully healed, plays not fully healed, etc.) to determine how happy I would be to hear that he's playing or not playing. Having calculated expected utilities for both "plays" and "doesn't play", I decide whether I'd prefer that he play or not.

Put another way, I can use Jeffrey's expected utility theory to determine whether I would consider it good news or bad were I to hear that my favorite player will be playing tonight. And I can do so whether or not I have *any* influence on the truth of that proposition. Jeffrey's theory is sometimes described as calculating the "news value" of a proposition.

Even for propositions describing our own acts, Jeffrey's expected utility calculation assesses news value. I might be given a choice between a sure $1 and a 50-50 chance of $2.02. I would use my credences and utility function to determine expected values for each act, then declare which option I preferred. But notice that this calculation would go exactly the same if instead

of my selecting among the options, someone else was selecting on my behalf. If my utility function assigns declining marginal utility to money, I might prefer just as much that someone else pick the sure dollar for me as I would prefer picking that option for myself. What's ultimately being compared are the proposition *that I receive a sure dollar* and the proposition *that I receive whatever payoff results from a particular gamble*. Whether I have the ability to make one of those propositions true rather than the other is irrelevant to Jeffrey's preference calculations.

7.3.1 Newcomb's Problem

Jeffrey's attention to news value irrespective of agency leads him into trouble with Newcomb's Problem. This problem was introduced to philosophy by Robert Nozick, who attributed its construction to the physicist William Newcomb. Here's how Nozick introduced the problem:

> Suppose a being in whose power to predict your choices you have enormous confidence. (One might tell a science-fiction story about a being from another planet, with an advanced technology and science, who you know to be friendly, etc.) You know that this being has often correctly predicted your choices in the past (and has never, so far as you know, made an incorrect prediction about your choices), and furthermore you know that this being has often correctly predicted the choices of other people, many of whom are similar to you, in the particular situation to be described below. One might tell a longer story, but all this leads you to believe that almost certainly this being's prediction about your choice in the situation to be discussed will be correct.
>
> There are two boxes. [The first box] contains $1,000. [The second box] contains either $1,000,000, or nothing. . . . You have a choice between two actions: (1) taking what is in both boxes (2) taking only what is in the second box.
>
> Furthermore, and you know this, the being knows that you know this, and so on:
>
> (I) If the being predicts you will take what is in both boxes, he does not put the $1,000,000 in the second box.
>
> (II) If the being predicts you will take only what is in the second box, he does put the $1,000,000 in the second box.
>
> The situation is as follows. First the being makes its prediction. Then it puts the $1,000,000 in the second box, or does not, depending upon what it has predicted. Then you make your choice. What do you do? (1969, pp. 114–5)

Historically, Newcomb's Problem prompted the development of a new kind of decision theory, now known as Causal Decision Theory (sometimes just "CDT"). At the time of Nozick's discussion, extant decision theories (such as Jeffrey's) seemed to recommend taking just one box in Newcomb's Problem (so-called "one-boxing"). But many philosophers thought two-boxing was the rational action.[10] Here's why: By the time you make your decision, the being has already made its prediction and taken its action. So the money is already either in the second box, or it's not—nothing you decide can affect whether the money is there. However much money is in the second box, you're going to get more money ($1,000 more) if you take both boxes. So you should two-box.

I've quoted Nozick's original presentation of the problem because in the great literature that has since grown up around Newcomb, there is often debate about what exactly counts as "a Newcomb Problem". Does it matter if the predictor is *perfect* at making predictions, or if the agent is *certain* that the prediction will be correct? Does it matter *how* the predictor makes its predictions, and whether backward causation (some sort of information fed backwards from the future) is involved? Perhaps more importantly, who *cares* about such a strange and fanciful problem?

But our purpose is not generalized Newcombology—we want to understand why Newcomb's Problem spurred the development of Causal Decision Theory. That can be understood by working with just one version of the problem. Or better yet, it can be understood by working with a kind of problem that comes up in everyday life, and is much less fanciful:

> I'm standing at the bar, trying to decide whether to order a third appletini. Drinking a third appletini is the kind of act much more typical of people with addictive personalities. People with addictive personalities also tend to become smokers. I'd kind of like to have another drink, but I *really* don't want to become a smoker (smoking causes lung-cancer, is increasingly frowned-upon in my social circle, etc.). So I shouldn't order that next appletini.

Let's work through the reasoning here on decision-theoretic grounds. First, stipulate that I have the following utility table:

	smoker	non
third appletini	−99	1
no more	−100	0

Ordering the third appletini is a dominant act. But dominance should dictate preference only when acts and states are independent, and my concern here is that they're not. My credence distribution has the following features (with A, S, and P representing the propositions that I order the appletini, that I become a smoker, and that I have an addictive personality, respectively):

$$\mathrm{cr}(S \,|\, P) > \mathrm{cr}(S \,|\, {\sim}P) \tag{7.10}$$
$$\mathrm{cr}(P \,|\, A) > \mathrm{cr}(P \,|\, {\sim}A) \tag{7.11}$$

I'm more confident I'll become a smoker if I have an addictive personality than if I don't. And having that third appletini is a positive indication that I have an addictive personality. Combining these two equations (and making a couple more assumptions I won't bother spelling out), we get:

$$\mathrm{cr}(S \,|\, A) > \mathrm{cr}(S \,|\, {\sim}A) \tag{7.12}$$

From my point of view, ordering the third appletini is positively correlated with becoming a smoker. Looking back at the utility table, I do not consider the states listed along the top to be probabilistically independent of the acts along the side. Now I calculate my Jeffrey expected utilities for the two acts:

$$\begin{aligned} \mathrm{EU}_{\mathrm{EDT}}(A) &= -99 \cdot \mathrm{cr}(S \,|\, A) + 1 \cdot \mathrm{cr}({\sim}S \,|\, A) \\ \mathrm{EU}_{\mathrm{EDT}}({\sim}A) &= -100 \cdot \mathrm{cr}(S \,|\, {\sim}A) + 0 \cdot \mathrm{cr}({\sim}S \,|\, {\sim}A) \end{aligned} \tag{7.13}$$

Looking at these equations, you might think that A receives the higher expected utility. But I assign a considerably higher value to $\mathrm{cr}(S \,|\, A)$ than $\mathrm{cr}(S \,|\, {\sim}A)$, so the -99 in the top equation is multiplied by a significantly larger quantity than the -100 in the bottom equation. Assuming the correlation between S and A is strong enough, ${\sim}A$ receives the better expected utility and I prefer to perform ${\sim}A$.

But this is all wrong! Whether I have an addictive personality is (let's say) determined by genetic factors, not anything I could possibly affect at this point in my life. The die is cast (so to speak); I either have an addictive personality or I don't; it's already determined (in some sense) whether an addictive personality is going to lead me to become a smoker. Nothing

cr : subjective prob. (credence)

about this appletini—whether I order it or not—is going to change that. So I might as well enjoy the drink.

Assuming the reasoning in the previous paragraph is correct, it's an interesting question why Jeffrey's decision theory yields the wrong result. The answer is that on Jeffrey's theory ordering the appletini gets graded down because it would be bad news about my future. If I order the drink, that's evidence that I have an addictive personality (as indicated in Equation (7.11)), which is unfortunate because of its potential consequences for becoming a smoker. I expect a world in which I order that drink to be a worse world than a world in which I don't, and this is reflected in the $\mathrm{EU}_{\mathrm{EDT}}$ calculation. Jeffrey's theory assesses the act of ordering a third appletini not in terms of the consequences it will *cause* to come about, but instead in terms of the consequences it provides *evidence* will come about. For this reason Jeffrey's theory is described as an Evidential Decision Theory (or "EDT").

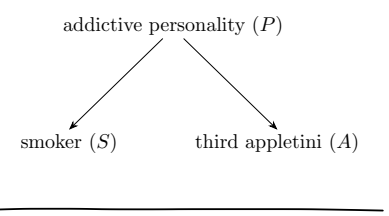
The trouble with Evidential Decision Theory is that an agent's performing an act may be *evidence* of a consequence that it's too late for her to *cause* (or *prevent*). Even though the act indicates the consequence, it seems irrational to factor the value of that consequence into a decision about whether to peform the act. As Skyrms (1980a, p. 129) puts it, my not having the third drink in order to avoiding becoming a smoker would be "a futile attempt to manipulate the cause by suppressing its symptoms." In making decisions we should attend to what we can control—to the causal consequences of our acts. Weirich writes,

> Deliberations should attend to an act's causal influence on a state rather than an act's evidence for a state. A good decision aims to produce a good outcome rather than evidence of a good outcome. It aims for the good and not just signs of the good. Often efficacy and auspiciousness go hand in hand. When they come apart, an agent should perform an efficacious act rather than an auspicious act. (2012)

7.3.2 A causal approach

The causal structure of our third drink example is depicted in Figure 7.1. As we saw in Chapter 3, correlation often indicates causation—but not *always*. Propositions on the tines of a causal fork will be probabilistically correlated even though neither causes the other. This accounts for A's being relevant to S on my credence function (Equation (7.12)) even though my ordering the third appletini has no causal influence on whether I'll become a smoker.

Figure 7.1: Third drink causal fork



The causally spurious correlation in my credences affects Jeffrey's expected utility calculation because that calculation works with credences in states conditional on acts ($\mathrm{cr}(S_i \mid A)$). Jeffrey replaced Savage's $\mathrm{cr}(S_i)$ with this conditional expression to track dependencies between states and acts. The Causal Decision Theorist responds that while credal correlation is a kind of probabilistic dependence, it may fail to track the causal dependences on which preferences should be based. So the Causal Decision Theorist's valuation function is:

$$\mathrm{EU}_{\mathrm{CDT}}(A) = \mathrm{u}(A \,\&\, S_1)\cdot\mathrm{cr}(A \mathrel{\square\!\!\rightarrow} S_1) + \mathrm{u}(A \,\&\, S_2)\cdot\mathrm{cr}(A \mathrel{\square\!\!\rightarrow} S_2) + \ldots + \mathrm{u}(A \,\&\, S_n)\cdot\mathrm{cr}(A \mathrel{\square\!\!\rightarrow} S_n) \tag{7.14}$$

Here $A \mathrel{\square\!\!\rightarrow} S$ represents the subjunctive conditional "If the agent were to perform act A, state S would occur."[11] Causal Decision Theory uses such conditionals to track causal relations in the world.[12] Of course, an agent may be uncertain what consequences a given act A would cause. So $\mathrm{EU}_{\mathrm{CDT}}$ looks across the partition of states $S_1, \ldots, S_n$ and invokes the agent's credence that A would cause any particular given S_i.

For many decision problems, Causal Decision Theory yields the same results as Evidential Decision Theory. In Jeffrey's wine example, it's plausible that

$$\mathrm{cr}(\text{chicken} \mid \text{white}) = \mathrm{cr}(\text{white} \mathrel{\square\!\!\rightarrow} \text{chicken}) = 0.75 \tag{7.15}$$

The guest's credence that chicken is served on the condition that she brings white wine is equal to her credence that if she were to bring white, chicken would be served. So one may be substituted for the other in expected utility calculations, and CDT's evaluations turn out the same as Jeffrey's.

But when conditional credences fail to track causal relations (as in cases with causal forks), the two theories may yield different results. This is in part due to their differing notions of independence. EDT treats act A and state S as independent when they are *probabilistically* independent relative to the agent's credence function. CDT focuses on whether the agent takes A and S to be *causally* independent, which occurs just when

$$\mathrm{cr}(A \mathrel{\square\!\!\rightarrow} S) = \mathrm{cr}(S) \tag{7.16}$$

When A has no causal influence on S, the agent's credence that S will occur if she performs A is just her credence that S will occur. In the third drink example my ordering another appletini may be evidence that I'll become a smoker, but it has no causal bearing on whether I take up smoking. So from a Causal Decision Theory point of view, the acts and states in that problem are independent. When acts and states are independent, dominance reasoning is appropriate, so I should prefer the dominant act and order that third appletini.

Now we can return to a version of the Newcomb Problem that distinguishes Causal from Evidential Decision Theory. Suppose that the "being" in Nozick's story makes its prediction by analyzing your brain state prior to your making the decision and applying a complex neuro-psychological theory. The being's track record makes you 99% confident that its predictions will be correct. And to simplify matters, let's suppose you assign exactly 1 util to each dollar, no matter how many dollars you already have. Then your utility and credence matrices for the problem are:

Utilities

	P_1	P_2
T_1	1,000,000	0
T_2	1,001,000	1,000

Credences

	P_1	P_2
T_1	0.99	0.01
T_2	0.01	0.99

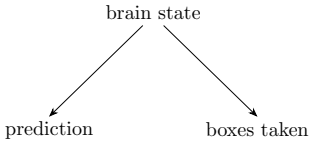
where T_1 and T_2 represent the acts of taking one box or two boxes (respectively), and P_1 and P_2 represent the states of what the being predicted.

Jeffrey calculates expected values for the acts as follows:

$$\begin{aligned}\mathrm{EU}_{\mathrm{EDT}}(T_1) &= \mathrm{u}(T_1 \,\&\, P_1)\cdot\mathrm{cr}(P_1 \mid T_1) + \mathrm{u}(T_1 \,\&\, P_2)\cdot\mathrm{cr}(P_2 \mid T_1) = 990{,}000\\ \mathrm{EU}_{\mathrm{EDT}}(T_2) &= \mathrm{u}(T_2 \,\&\, P_1)\cdot\mathrm{cr}(P_1 \mid T_2) + \mathrm{u}(T_2 \,\&\, P_2)\cdot\mathrm{cr}(P_2 \mid T_2) = 11{,}000\end{aligned} \tag{7.17}$$

So Evidential Decision Theory recommends one-boxing. Yet we can see from Figure 7.2 that this version of the Newcomb Problem contains a causal fork;

Figure 7.2: Newcomb Problem causal fork



the being's prediction is based on your brain state, which also has a causal influence on the number of boxes you take. This should make us suspicious of EDT's recommendations. The agent's act and the being's prediction are probabilistically correlated in the agent's credences, as the credence table reveals. But that's not because the number of boxes taken has any causal influence on the prediction.

Causal Decision Theory calculates expected utilities in the example like this:

$$\begin{aligned} \mathrm{EU}_{\mathrm{CDT}}(T_1) &= \mathrm{u}(T_1 \,\&\, P_1) \cdot \mathrm{cr}(T_1 \mathbin{\square\!\!\rightarrow} P_1) + \mathrm{u}(T_1 \,\&\, P_2) \cdot \mathrm{cr}(T_1 \mathbin{\square\!\!\rightarrow} P_2) \\ &= 1,000,000 \cdot \mathrm{cr}(T_1 \mathbin{\square\!\!\rightarrow} P_1) + 0 \cdot \mathrm{cr}(T_1 \mathbin{\square\!\!\rightarrow} P_2) \\ \\ \mathrm{EU}_{\mathrm{CDT}}(T_2) &= \mathrm{u}(T_2 \,\&\, P_1) \cdot \mathrm{cr}(T_2 \mathbin{\square\!\!\rightarrow} P_1) + \mathrm{u}(T_2 \,\&\, P_2) \cdot \mathrm{cr}(T_2 \mathbin{\square\!\!\rightarrow} P_2) \\ &= 1,001,000 \cdot \mathrm{cr}(T_2 \mathbin{\square\!\!\rightarrow} P_1) + 1,000 \cdot \mathrm{cr}(T_2 \mathbin{\square\!\!\rightarrow} P_2) \end{aligned} \tag{7.18}$$

It doesn't matter what particular values the credences in these expressions take, because the act has no causal influence on the prediction. That is,

$$\mathrm{cr}(T_1 \mathbin{\square\!\!\rightarrow} P_1) = \mathrm{cr}(P_1) = \mathrm{cr}(T_2 \mathbin{\square\!\!\rightarrow} P_1) \tag{7.19}$$

and

$$\mathrm{cr}(T_1 \mathbin{\square\!\!\rightarrow} P_2) = \mathrm{cr}(P_2) = \mathrm{cr}(T_2 \mathbin{\square\!\!\rightarrow} P_2) \tag{7.20}$$

With these causal independencies in mind, you can tell by inspection of Equation (7.18) that $\mathrm{EU}_{\mathrm{CDT}}(T_2)$ will be greater than $\mathrm{EU}_{\mathrm{CDT}}(T_1)$, and Causal Decision Theory endorses two-boxing.