Decision
2017, Vol. 4, No. 2, 67–86
© 2016 American Psychological Association
2325-9965/17/$12.00 http://dx.doi.org/10.1037/dec0000053

Structured Representations of Utility in Combinatorial Domains

Samuel J. Gershman
Harvard University

Jonathan Malmaud and
Joshua B. Tenenbaum
Massachusetts Institute of Technology

People can judge whether they will enjoy dishes like waffles with horseradish cream sauce or broccoli ice cream even if they have never tried them. What representations and computations support reasoning in such situations? We develop a theory of decision making in combinatorial domains. Its central claim is that utility functions can be compositionally structured: The utility of a combination is a function of its constituents' utilities and the rules for combining them. Utilities are induced from experience by probabilistic reasoning over the structured space of utility functions. In a series of experiments, we show how this theory can capture human evaluations of novel food combinations. We first show that the theory quantitatively predicts evaluations of novel food combinations. We then report more strongly controlled experiments (using unfamiliar foods) that rule out several alternative theories. Taken together, these experiments demonstrate how compositionally structured representations of utility can support decision making in combinatorial domains.

Keywords: Bayesian inference, utility theory, compositionality

Supplemental materials: http://dx.doi.org/10.1037/dec0000053.supp

This document is copyrighted by the American Psychological Association or one of its allied publishers.
This article is intended solely for the personal use of the individual user and is not to be disseminated broadly.

Visitors to the restaurant Dirt Candy in New York have the opportunity to try such dishes as Kimchi doughnuts, waffles with horseradish cream sauce, and broccoli ice cream. How do visitors judge whether these novel combinations will taste good? A similar question arises in many other domains: Will these pants go with this shirt? Which couch would look best in my living room? Who should I invite to my dinner party? Despite the ubiquity of such questions, surprisingly little is known about how people make decisions in combinatorial domains. In this article, we develop a normative theory of combinatorial decision making and test its predictions experimentally.

Like many theories of decision making, we assume that preference is determined by a utility function that assigns scalar utilities to objects in the choice set. The central claim of our theory is that utility functions can be compositionally structured: The utility of a combination is a function of its constituents' utilities and the rules for combining them. Compositionality endows an agent with the ability to evaluate (assign utility to) a virtually infinite number of combinations, a property known as *productivity*. Compositionality also guarantees *systematicity*: If an agent can evaluate a particular combination, then they can also evaluate a structurally similar combination (e.g., the ability to evaluate "tuna sandwich" and "chicken salad" implies the ability to evaluate "chicken sandwich" and "tuna salad").

Our theory attempts to situate decision making within the larger framework of the *language of thought* (Fodor, 1975), which asserts that cognition is a compositional system, building

This article was published Online First April 7, 2016.
Samuel J. Gershman, Department of Psychology and Center for Brain Science, Harvard University; Jonathan Malmaud and Joshua B. Tenenbaum, Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology.
We are grateful to Drazen Prelec for helpful discussions. This research was supported by ARO MURI contract W911NF-08-1–0242, the Center for Brains, Minds and Machines (CBMM), funded by NSF STC Award CCF-1231216, and the Office of the Director of National Intelligence (ODNI), and a postdoctoral fellowship from the MIT Intelligence Initiative (SJG).
Correspondence concerning this article should be addressed to Samuel J. Gershman, Department of Psychology and Center for Brain Science, Harvard University, Northwest Building, 52 Oxford Street, Room 295.05, Cambridge, MA 02138. E-mail: gershman@fas.harvard.edu

complex representations out of simpler ones through the application of composition laws. More recent versions of this framework have formalized a *probabilistic* language of thought (Goodman, Tenenbaum, Feldman, & Griffiths, 2008; Goodman, Ullman, & Tenenbaum, 2011; Kemp, Goodman, & Tenenbaum, 2008; Piantadosi, Tenenbaum, & Goodman, 2012), which accommodates uncertainty by allowing complex representations to be probabilistic functions of their constituents. Probabilistic inference over this language of thought can capture both the compositional structure of cognitive representations and the graded nature of learning and reasoning. Our goal is to show how these same ideas can be used to understand decision making in combinatorial domains.

Below, we formalize our theory, showing how it extends classical linear theories to combinatorial domains. Classical linear theories assume that objects are encoded by a set of features and conjunctions of features; utility is modeled as a linear function of these features and conjunctions. This approach ignores the underlying object structure, encoding conjunctions of features even when these features belong to different object parts. In contrast, our compositional utility theory respects object structure, defining a compositional procedure for encoding structure-sensitive features. In particular, we develop a theory of utility functions defined over tree structures that represent the part–whole structure of objects. This representation is combined with a probabilistic inference engine that enables an agent to reason inductively about objects in combinatorial domains. Our goal is to take a step toward a more realistic theory of utility by combining the power of classical linear theories with a more sophisticated object representation and principles of probabilistic reasoning. As we elaborate below, our theory is similar in some respects to earlier tree-structured utility models (Gorman, 1968; Strotz, 1957; Tversky & Sattath, 1979), but goes beyond them by showing how the parameters governing the utility function can be learned from experience in a manner consistent with normative inductive principles.

We present a series of experiments designed to validate the theory. First, we asked participants to rate various combinations of familiar foods. We show that our model quantitatively predicts these ratings better than classical linear theories. Second, we present results from controlled experiments (using unfamiliar foods) showing that human reasoning about utilities is sensitive to object structure in a way that is inconsistent with classical linear theories.

Theoretical Background

Because our work builds upon classical theories of utility, we begin by discussing the historical background. The relevant literature is vast and complex, so we must satisfy ourselves with a selective review. We will not cover decision making under uncertainty, focusing instead on issues of representation.

A cardinal utility function $u : \chi \mapsto \mathbb{R}$ is a mapping from an object (or outcome) space χ to the real numbers. According to classical utility theory (Von Neumann & Morgenstern, 1947), an agent prefers $x \in \chi$ to $x' \in \chi$ if, and only if, $u(x) > u(x')$. In multiattribute utility theory (Keeney & Raiffa, 1976), objects are represented by a collection of D attributes, $x = \{x_1, \ldots, x_D\}$, and the object space becomes a Cartesian product: $\chi = \chi_1 \times \ldots \times \chi_D$.

Much of the research in multiattribute utility theory has focused on establishing necessary and sufficient conditions for utility functions to be factorizable into local functions $\{u_d(x_d)\}$ over individual attributes:

$$u(x) = f[u_1(x_1), \ \ldots, \ u_D(x_D)], \tag{1}$$

where $f[\cdot]$ is a composition function. The most well-studied form of factorization is the additive (linear) utility function (e.g., Krantz, Luce, Suppes, & Tversky, 1971; Wakker, 1989):

$$u(x) = \sum_{d=1}^{D} u_d(x_d). \tag{2}$$

Econometric models, most notably the multinomial logit model (McFadden, 1973), frequently assume additivity because of the tractability it confers for modeling preferences: The number of parameters is linear in the number of attributes, rather than exponential in the general case. Additivity has also been employed in other related domains, such as value function approximation in reinforcement learning (Sutton & Barto, 1998) and outcome prediction in classical conditioning (Rescorla & Wagner, 1972).

This document is copyrighted by the American Psychological Association or one of its allied publishers.
This article is intended solely for the personal use of the individual user and is not to be disseminated broadly.

This document is copyrighted by the American Psychological Association or one of its allied publishers.
This article is intended solely for the personal use of the individual user and is not to be disseminated broadly.

Because additivity was recognized to be too restrictive for some applications, Fishburn (1967) proposed a generalized additive formulation (see also Bacchus & Grove, 1995; Braziunas & Boutilier, 2009), in which the utility function decomposes into a sum of M factors (attribute subsets):

$$u(x) = \sum_{m=1}^{M} u_m(x_m), \tag{3}$$

where $x_m \subseteq x$. The compositional utility theory that we present in the next section can be seen as a special case of generalized additive utility, in which the attribute factors are tailored to a particular form of object representation—namely, a tree structure.

The idea that utility functions or choice processes can be defined over tree structures has several precedents. Tversky and Sattath (1979) proposed a model called *preference tree*, according to which choices are made by a hierarchical elimination process. An option is represented by a number of "aspects" (i.e., attributes) arranged into a tree, and each aspect is assigned some probability. At each stage of the elimination process, a single aspect is probabilistically selected and all options that do not have that aspect are eliminated. The process continues down the tree (starting at the node corresponding to the most recently selected aspect) until only a single option remains. Like its more general predecessor, *elimination by aspects* (Tversky, 1972), preference tree can capture similarity and context effects in choice behavior. However, it is much more parsimonious, with $2D$–2 rather than 2^D–2 parameters.

Strotz (1957) and Gorman (1968) explored utility functions that are decomposable into a tree structure. This can be viewed as a form of factorization: Utility is factorized into *branch utility functions* defined over mutually exclusive and exhaustive subsets of the attributes. The key implication of this factorization is that the marginal rate of substitution between two attributes in the same branch is independent of attributes in other branches. In other words, the rate at which an agent is willing to give up some amount of one attribute (e.g., bananas) in exchange for another attribute (e.g., oranges) in the same branch, while keeping utility fixed, does not depend on the amount of any attribute in other branches (e.g., shirts, books).

Our goal is to combine the key advantages of the classical linear multiattribute models of utility with the tree-based approaches. The linear multiattribute models are generative, in that they can generalize to combinations that have never been experienced before—an important asset for any real theory of value. Furthermore, the parameters of linear models can be learned using probabilistic or reinforcement learning algorithms (Rasmussen & Williams, 2006; Sutton & Barto, 1998). However, these models, as traditionally formulated, do not deal with compositional or hierarchical structure needed to evaluate complex prospects in the real world. The tree models described above have the latter property, but they do not capture how utility functions are learned from experience. Our goal here is to produce a useful, practical mathematical model for utility that has these two key properties: It is capable of handling complex objects compositionally, and it can support rational inferences about utilities from sparse data—just one or a few experiences—using principles of probabilistic reasoning.

Because the tree-based models like Tversky and Sattath (1979) are descriptive models of choice that are not designed to explain how the parameters governing utility assessment are learned from experience, our experiments here focus on comparing our hierarchical model with linear multiattribute models and, specifically, probabilistic inference-based versions of linear theories that correspond to special cases of our approach. This also allows us to show, in a well-controlled way, the added value of hierarchical structure in our framework for predicting choice and value.

Compositional Utility Theory

In this section, we introduce compositional utility theory. We first define a representation of objects in combinatorial domains, and then describe how utility functions over such objects can be induced from experience. This model is mathematically equivalent to a linear parameterization of the utility function with a particular set of structure-sensitive features, allowing us to draw connections to a family of alternative models. Finally, we show how appropriate representational constraints can generate utility functions with compositional properties.

Object Representation

Our focus is on composite objects (e.g., broccoli ice cream is composed of broccoli and ice cream), in which the constituent objects may themselves be composite. We refer to objects that cannot be decomposed further into other objects as *primitives*, and each primitive is associated with a set of attributes (e.g., vegetable, sweet, dairy). Formally, a composite object can be represented as a tree whose root node is the complete composite object and whose leaf nodes are primitive attributes obtained by recursively expanding composite objects into their constituents. An example is shown in Figure 1A. Trees can be equivalently represented using set notation:

$$\begin{aligned}\text{broccoli ice cream} &= \{\text{broccoli, ice cream}\}\\ &= \{\{\text{vegetable, savory}\},\\ &\quad\ \{\text{dairy, sweet}\}\}.\end{aligned}$$

Although this object representation is quite simple, it captures the important part–whole structure of objects that has been neglected in studies of decision making.

Utility Representation

Before proceeding, it is worth noting one way in which our utility theory is different from standard treatments. We view utility as *experienced* in the sense that it encodes hedonic pleasure or pain (Bentham, 1879; Kahneman, Wakker, & Sarin, 1997). As such, the agent only samples the utilities of experienced (i.e., chosen) objects. The utilities of other objects must be inferred. The *decision utility* of an object reflects its role in determining choice behavior, which, in our theory, entails an inferential computation using noisy samples from memory (for related ideas, see Gilboa & Schmeidler, 1995; Gonzalez, Lerch, & Lebiere, 2003; Hertwig & Erev, 2009; Stewart, Chater, & Brown, 2006).

When an agent experiences object x, its utility $u(x)$ is stored in memory as r:

$$r = u(x) + \epsilon, \tag{4}$$

where $\epsilon \sim \mathcal{N}(0, \tau)$ is a noise term that captures corruption in memory; larger values of τ correspond to greater corruption.[1] Before sampling any objects, the agent's probabilistic beliefs about the latent utility function are modeled using a Gaussian process (GP; Rasmussen & Williams, 2006): $u \sim \mathrm{GP}(m, k)$, where $m(x) = \mathbb{E}[u(x)]$ is the mean function, which we assume for simplicity to be zero for all x, and k is the covariance (or *kernel*) function:

$$k(x, x') = \mathbb{E}[(u(x) - m(x))(u(x') - m(x'))]. \tag{5}$$

The kernel function determines the smoothness of the utility function over the object space χ. Intuitively, $k(x, x')$ expresses the similarity between x and x', such that similar objects will tend to have similar utilities.

We require a covariance function that respects the inherent compositionality of objects: Objects with similar trees should have similar utilities. We therefore adopt a covariance function known as the *subtree kernel* (Moschitti, 2006; Vishwanathan & Smola, 2002), which counts the number of common subtrees whose leaf nodes are primitive attributes:

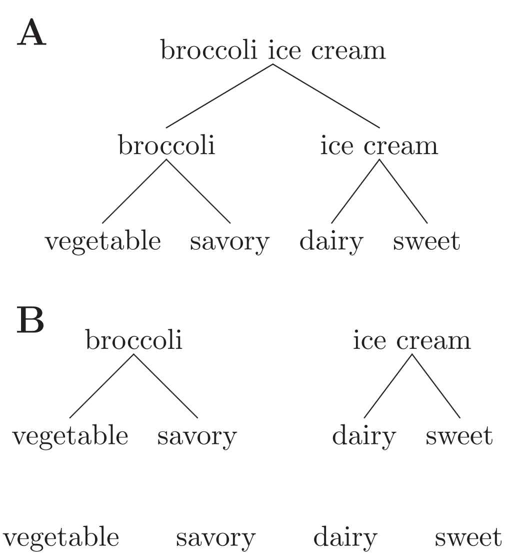


Figure 1. Example of tree-structured object representation. (A) Simplified representation of broccoli ice cream. (B) Subtrees of broccoli ice cream.

[1] From a neural perspective, we do not require that the physical substrate of the memory is corrupted; corruption could also occur at the time of retrieval.

This document is copyrighted by the American Psychological Association or one of its allied publishers.
This article is intended solely for the personal use of the individual user and is not to be disseminated broadly.

$$k(x, x') = \sum_{t \in x} \sum_{t' \in x'} C(t, t'), \tag{6}$$

where t is a node (object or attribute) in the object parse and $C(t, t') = 1$ if the subtrees rooted at t and t' are identical (0 otherwise). Using this covariance function, objects are treated as similar to the extent that they share common subtrees (some examples of subtrees are shown in Figure 1B). We will refer to the GP model using the subtree kernel simply as the "tree model."

Probabilistic Inference and Decision Making

Given a new object x_* and memory traces of N experiences, $D = \{x_n, r_n\}_{n=1}^N$, the problem facing the agent is to estimate $u(x_*)$. The posterior distribution over $u(x_*)$ is given by

$$P(u(x_*) | \mathcal{D}) = \mathcal{N}(u(x_*); \bar{u}_*, \sigma_*^2), \tag{7}$$

with mean and variance

$$\bar{u}_* = \mathbf{k}_*^\top (\mathbf{K} + \tau \mathbf{I})^{-1} \mathbf{r} \tag{8}$$

$$\sigma_*^2 = k(x_*, x_*) - \mathbf{k}_*^\top (\mathbf{K} + \tau \mathbf{I})^{-1} \mathbf{k}_*, \tag{9}$$

where $r = [r_1, \ldots, r_n]^\top$, $K_{ij} = k(x_i, x_j)$ and $k_* = [k(x_*, x_1), \ldots, k(x_*, x_N)]^\top$. Note that we have distinguished the kernel function $k(x, x')$ from finite-dimensional matrices and vectors ($\mathbf{K}$ and $\mathbf{k}_*$) constructed by evaluating the kernel function at specific pairs of points. The posterior mean $\bar{u}_*$ represents the agent's "affective forecast" (Kahneman & Snell, 1992; Wilson & Gilbert, 2005) about the utility of x_*.

To gain some intuition for these equations, we can express the posterior mean as a linear combination of experienced utilities (Rasmussen & Williams, 2006):

$$\bar{u}_* = \sum_{n=1}^{N} \alpha_n r_n, \tag{10}$$

where $\alpha = k_*^\top (K + \tau I)^{-1}$. Thus, α_n can be seen as weighting an individual experience, using the similarity between the new object x_* and object x_n. In this sense, the utility prediction is similar to an exemplar model, as pointed out by Lucas, Griffiths, Williams, and Kalish (2015) in the context of function learning.

The posterior distribution can be used to compute the probability that x_* has a higher utility than another object, x_0:

$$P(u(x_*) > u(x_0)) = \Phi\big(0, \bar{u}_* - \bar{u}_0, \sigma_*^2 + \sigma_0^2\big), \tag{11}$$

where Φ is the Gaussian cumulative distribution function. This form of choice probability function is similar to several other models of multiattribute choice (see Carroll & De Soete, 1991). We will use this probability to model participants' choices in some of the experiments reported below. Intuitively, the probability of choosing x_* increases as its mean utility becomes larger than the mean utility of x_0, and the slope of this increase is inversely proportional to the posterior variance of the two distributions. The more the distributions overlap, the less sensitive is the choice probability function to differences in mean utility.

Relationship to Linear Models

It can be shown that the subtree kernel model is equivalent to a linear parameterization of the utility function (see Rasmussen & Williams, 2006):

$$u(x) = \sum_d w_d f_d(x), \tag{12}$$

where $f_d(x)$ counts the number of occurrences of subtree d in object x, and w_d is a scalar weight, randomly drawn from a Gaussian with zero mean and unit variance. The subtree kernel is derived from an inner product between feature vectors:

$$k(x, x') = \sum_d f_d(x) f_d(x'). \tag{13}$$

This equivalence gives us insight into how the subtree kernel model is related to classic multiattribute utility models, which posit a linear utility function of the same form as Equation 12, but where $f_d(x)$ encodes an attribute of x (Keeney & Raiffa, 1976). This model can be derived from the subtree kernel when objects are rooted at a single primitive. In other words, linear multiattribute models are obtained as a

This document is copyrighted by the American Psychological Association or one of its allied publishers.
This article is intended solely for the personal use of the individual user and is not to be disseminated broadly.

This document is copyrighted by the American Psychological Association or one of its allied publishers.
This article is intended solely for the personal use of the individual user and is not to be disseminated broadly.

special case of the subtree kernel model without hierarchical structure; objects are simply collections of nondecomposable attributes.

In modeling our experimental data, we will consider a broader range of linear models, which differ only in their choice of features, $\{f_d(x)\}$.[2] These models can be divided into *counter* models, where $f_d(x)$ counts the number of times a particular attribute occurs across all primitive objects, and *detector* models, where the counts are thresholded to $\{0, 1\}$. These models are first-order, in the sense that features only count (or detect) single attribute occurrences. We can also consider higher order models with features that count (or detect) attribute co-occurrences (in practice we only consider second-order models). For all the models we consider, we apply the same probabilistic inference and decision making machinery described above.

It is useful to think of the subtree kernel as a particular kind of higher order counter model, which can encode co-occurrences of arbitrarily high order, but only when attributes co-occur in the same subtree. In the example given above, the subtree kernel would have a feature corresponding to a third-order conjunction (banana ice cream, which has three unique attributes at the leaves of the tree). However, it would not have a feature corresponding to the second-order conjunction {fruit, dairy}, because these never co-occur in the same subtree, except as part of larger conjunctions. Thus, the subtree kernel is built out of feature conjunctions just like other linear models, but the conjunctions it encodes are dictated by the underlying object structure. We will show experimentally that this is a central property of human reasoning about utilities. From a computational perspective, the connection between kernels and linear models implies that the same algorithmic machinery underlying standard multiattribute decision making can support decision making with tree-structured utility functions.

To gain some further intuition about how the linear parameterization distinguishes different utility representations, consider the following example (a simplified version of Choice Problem 1 in Experiment 2) involving three objects: a is sweet but not salty, b is sweet and salty, and c is both sweet and salty. If $r_a = 1$, then a first-order counter model will assign a weight (in the linear parameterization) of roughly 1 to the sweat feature, and 0 to the salty feature, with the reverse assignments for object b. In this example, the counter and detector models are identical because each attribute only occurs once in each object. The subtree model has an additional binary feature representing the occurrence of each object, and hence the object feature and the primitive features compete to predict utility, resulting in weights around 0.5. Now consider a choice between c and the composite $a + b$ (i.e., consuming two separate food objects together). These two options have the same primitive features, which means that the counter and detector models are indifferent between the two options (because these models predict utility as a linear combination of primitive features). Specifically, these models predict a utility of roughly 2 for both options. The subtree model posits that the appearance of objects a and b in the composite $a + b$ add their object-specific weights, resulting in a predicted utility of roughly 3 for $a + b$ but only 2 for c (because c is a novel object, its object-specific weight is 0). Thus, the subtree model predicts a preference for $a + b$ over c. We provide support for this prediction in Experiment 2.

Composition Laws

In what sense does the subtree kernel generate compositional properties of utility? Suppose that object x is a composite of objects a and b (which themselves may be composite). Then the subtree kernel has the following property:

$$k(x, x') = k(a, x') + k(b, x') + \sum_{t' \in x'} C(x, t'). \tag{14}$$

If x' does not contain x as a subtree, then the third term is equal to 0 and the kernel for the composite is a superposition of kernels for the constituents.

Kernel compositionality gives rise to utility compositionality. This is easiest to understand in the linear parameterization. Given memory traces D, the prediction for x_* is a linear function of the posterior mean weights, $\hat{\mathbf{w}}$:

[2] Note that a feature is more general than an attribute: The latter refers to a single property of a primitive object (e.g., sweet, salty, etc.), whereas a feature can be a function of multiple attributes across multiple objects.

$$\overline{u}(x_*) = \sum_d \hat{w}_d f_d(x_*) \tag{15}$$

$$\hat{\mathbf{w}} = \mathbf{f}^\top(\mathbf{K} + \tau\mathbf{I})^{-1}\mathbf{r}. \tag{16}$$

The rows of matrix $\mathbf{F}$ are feature vectors for objects 1 to N. Because the prior mean for the weights is 0, the posterior mean for w_d is also 0 unless subtree d has been encountered before. Using the example in the previous paragraph, let $\hat{\mathbf{w}}_a$ denote the vector of weights for all subtrees in object a, and let $\mathbf{f}_a$ denote the corresponding vector of features evaluated on object x. The utility of x can then be decomposed according to

$$\begin{aligned}\overline{u}(x) &= \hat{\mathbf{w}}_a\mathbf{f}_a^\top + \hat{\mathbf{w}}_b\mathbf{f}_b^\top + \hat{w}_x f_x(x)\\ &= \overline{u}(a) + \overline{u}(b) + \hat{w}_x f_x(x),\end{aligned}$$

where $\hat{\mathbf{w}}_x$ is the weight associated with the single subtree x. If x is novel (i.e., was never encountered as a subtree of another object), then the third term is equal to 0, and we see that the utility of a novel object is the superposition of its constituents' utilities. Thus, utilities also obey an additive composition law.

This analysis also gives insight into the conditions under which "the whole is more than the sum of its parts." If object x was previously encountered as a subtree of another object, then its corresponding feature will have a nonzero weight (assuming the experienced utility was nonzero). This property enables the model to capture nonlinear utility functions, such as when two foods are enjoyed in isolation but not in combination (e.g., salmon milkshake).

Experiments 1A to 1C

The goal of Experiments 1A–1C was to quantitatively compare the predictive accuracy of the tree model with several alternative models in a realistic domain. We chose the domain of food combinations because this task is common in everyday life. Experiment 1A used combinations of one, two, or three "simple" ingredients (see Table 1)—that is, ingredients that are directly decomposable into primitive attributes (of course, this is only a gross approximation of reality for the ingredients we consider). Each participant rated the desirability of combinations on a 0-to-9 scale. For each ingredient, we also collected judgments (from a separate group of participants) of 17 primitive attributes. These attributes were used as the input into several different utility models, including the tree model. We quantitatively compared the predictive accuracy of these models and provide some intuition for differences in performance. Experiment 1B extends these results to more complex foods consisting of combinations of combinations (e.g., a hamburger is—to a first approximation—a combination of bun and burger, each of which is decomposable into primitive attributes). Experiment 1C extended the results of Experiments 1B and 1C to a choice task.

Materials and Methods

Participants. Participants were recruited through the Amazon Mechanical Turk service. Experiments 1A and 1B consisted of an attribute survey and a desirability rating task, run in separate groups of participants; thus, there were four groups total across Experiments 1A and

Table 1
Examples of Low- and High-Rated Combinations of Three Ingredients in Experiment 1A

Low-rated	High-rated
mayonnaise + banana + eggs	garlic + tomato + salsa
mayonnaise + chili + raisins	mashed potato + cheddar cheese + mushrooms
mayonnaise + banana + cayenne	turkey + cayenne + black pepper
mayonnaise + chili + banana	black pepper + mashed potato + biscuits
mayonnaise + garlic + chocolate	mashed potato + cheddar cheese + tomato
sugar + green pepper + ice cream	black pepper + mashed potato + garlic
mayonnaise + chili + ice cream	turkey + black pepper + onion
mayonnaise + chili + maple syrup	black pepper + mashed potato + cheddar cheese
red wine + milk + orange zest	black pepper + mashed potato + onion
sugar + parsley + mushrooms	black pepper + mashed potato + salt

This document is copyrighted by the American Psychological Association or one of its allied publishers.
This article is intended solely for the personal use of the individual user and is not to be disseminated broadly.

This document is copyrighted by the American Psychological Association or one of its allied publishers.
This article is intended solely for the personal use of the individual user and is not to be disseminated broadly.

1B. In the attribute survey, participants ($n = 43$ for Experiment 1A and $n = 20$ for Experiment 1B) were paid $2.00 for completing the survey. In the desirability rating task, participants ($n = 43$ for Experiment 1A, $n = 20$ for Experiment 1B) were paid $3.00 for completing the survey. Nine participants in Experiment 1A were excluded because they displayed insufficient variance in their ratings for the model to be adequately estimated (greater than 60% of the items were given the same rating). Thirty-six participants in Experiment 1C did both a desirability rating task and a binary choice task. We determined sample sizes based on typical decision making studies with 20 to 40 participants.

Materials. We manually created a list of 44 ingredients, which included foods from a variety of genres. A set of 17 primitive attributes was used to model the data from Experiments 1A and 1B: savory, spicy, sweet, sour, bitter, fatty, healthy, crispy, squishy, creamy, tender, seafood, meat, dairy, fruit, vegetable, and starch. Note that participants in the desirability rating task were not shown these primitive attributes; they only entered our analyses through the construction of the covariance matrix **K**.

Procedure. All measures and conditions are described below. No measures or conditions were omitted.

Attribute survey. The survey was presented as a list of ingredients, each followed by the set of possible features, with a drop box next to each feature in which the participant could input a rating on a 1-to-10 scale. Each participant rated all 44 ingredients along the 17 attributes. The order of the ingredients and the order of the attributes were randomized across participants. We converted these ratings to binary (0, 1) by thresholding them at 5.

Desirability rating task. Participants in the desirability rating task were shown a combination of ingredients and asked to rate its desirability on a scale from 1 to 10. In Experiment 1A, each participant was shown the same set of 477 combinations. This included all 44 single ingredients, and a randomly selected subset of pairs and triplets. In Experiment 1B, all participants were shown the same set of 287 combinations, including "complex" foods, which were decomposable into other ingredients, such as hamburger and grilled cheese. In Experiment 1C, all participants were shown the same set of 400 combinations.

Binary choice task. Participants in the binary choice task were given a choice between two ingredient combinations. All participants made choices between the same set of 100 pairs.

Model-fitting and evaluation. We compared the tree model with four different linear models that varied in a 2×2 space: first-order detector models (detector-1), second-order detector models (detector-2), first-order counter models (counter-1) and second-order counter models (counter-2). These models (including the tree model) differ only in their feature representation. All the models form predictions in the same way, using the GP equations described above. We take the posterior mean (Equation 8) as our prediction of participants' desirability ratings, corrupted by Gaussian noise.[3] All the models have only a single free parameter, τ, which we fit using a coarse grid-search to maximize the log likelihood of participants' desirability ratings.

Fitting followed a leave-one-out cross-validation procedure, in which τ was fit to all the ratings except one, and the fitted model was used to make a prediction for the held-out rating; this procedure was repeated for all ratings and applied to each participant's data separately. Cross-validation provides us with an unbiased estimate of a model's generalization ability. In addition, we evaluated the models using group-wise Bayesian model comparison, as described in Stephan, Penny, Daunizeau, Moran, and Friston (2009), which quantifies model fit using the *exceedance probability*—the probability that a particular model is better than all the other models considered. This method assumes that each participant's data are generated by a single model (taken from the finite set considered here) and that the probability of selecting a particular model is a random variable. The selection probability is estimated using Bayesian inference, and the exceedance probability is computed from the resulting posterior over models.[4]

Results and Discussion

To get a visual sense of how the data and model predictions look, we applied the t-SNE algorithm (Van der Maaten & Hinton, 2008) to obtain a two-dimensional embedding of the

[3] Because desirability ratings are bounded, a Gaussian noise model is somewhat ill-suited, and could potentially be improved by using a cumulative probit model.

[4] For a standalone MATLAB implementation of this procedure, see https://github.com/sjgershm/mfit/blob/master/bms.m.

food combinations, using the tree kernel as a similarity metric. Figure 2 shows the embedding, with point size scaled by the average rating for each combination. A subset of combinations (all including "broccoli") is shown for illustration. As expected, combos with the same ingredients tended to cluster together, but there is spread both in the distribution of ratings and their location in the embedding space.

The estimated value of τ for Experiment 1A was 7.18; this estimate was fairly stable across training folds, and across the subsequent experiments, with values typically ranging between 6.5 and 7.5. Figure 3 shows the model comparison results for Experiment 1A. We found that the tree model had significantly lower root mean squared error on held-out data compared with its closest competitor, the detector-2 model, $t(33) = 2.07$, $p < .05$ (Figure 3A). The average correlation coefficient between the tree model predictions and empirical ratings was 0.24, significantly greater than 0, $t(33) = 7.17$, $p < .0001$. The correlation coefficients for the other models were 0.16 (detector-1), 0.21 (detector-2), 0.14 (counter-1), and 0.19 (counter-2). The tree model was also found to have the highest exceedance probability (Figure 3B), indicating that the tree model has the strongest statistical evidence across the group of participants. These results provide quantitative support for the assertion that human utility functions (at least for food judgments) rely upon a tree-structured representation.

What is the origin of this superior predictive performance? A hallmark of the tree model is that it explicitly encodes objects and uses them to structure its generalizations. This means it can learn that two objects with very similar attributes can have very different utilities, as long as the attributes belong to different objects. Put another way, object-encoding features can absorb residual variance in the utilities that are not captured by the attributes (or conjunctions of attributes). This idea is illustrated in Figure 4,

Figure 2. Embedding of tree model similarities. The size of each point is proportional to its average rating. Shown here are only triplet combinations, with all combinations involving "broccoli" highlighted in blue for illustration. See the online article for the color version of this figure.

This document is copyrighted by the American Psychological Association or one of its allied publishers.
This article is intended solely for the personal use of the individual user and is not to be disseminated broadly.

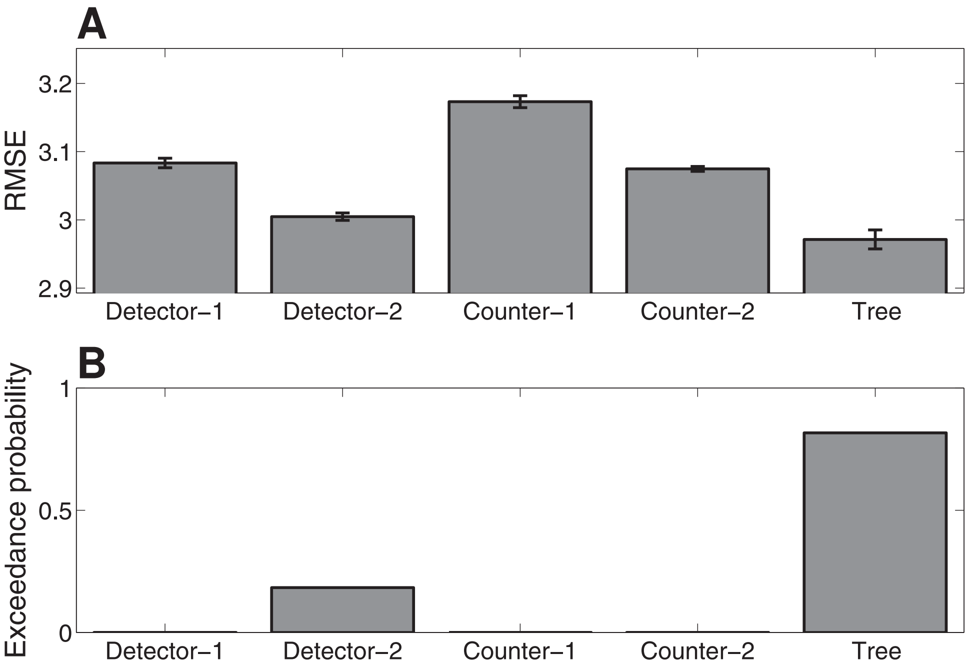


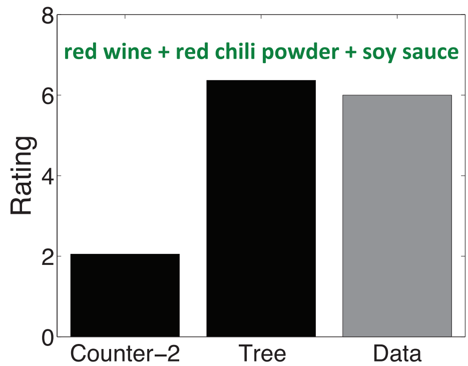
Figure 3. Model comparison: Experiment 1A. (A) Root mean squared error (RMSE) for each model's predictions, computed on held-out data using a leave-one-out cross-validation procedure. Lower values indicated better predictions. Error bars represent within-subject standard errors. (B) Exceedance probability for each model—the probability that the likelihood of a particular model is greater than all the other models (Stephan et al., 2009). Higher values indicate stronger model evidence.

in which the tree model predicts that a participant will moderately like the combination "red wine + chili powder + soy sauce," whereas the counter-2 model fails. Figure 4 (bottom) shows the six nearest neighbors of the target combination, as measured by the kernel function *k*. These combinations are sorted by decreasing utility (*r*). The tree kernel picks out combinations as highly similar that share objects with the target combination; in this case, these combinations happen to have relatively high utility. The counter-2 kernel, in contrast, lacks this "object bonus," preferring instead combinations that share similar feature conjunctions (which, in this case, happen to have slightly lower utility compared with the combinations picked out by the tree kernel).

In Experiment 1B, we sought to extend these results with more complex foods, consisting of higher level combinations, such as hamburger (bun + burger) and hamburger + chocolate milk (see Table 2 for examples of combinations of complex foods used in the experiment). Apart from these new combinations, the procedure was identical to Experiment 1A. Figure 5 shows the model comparison results for Experiment 1B, demonstrating once again that the tree model captures participants' desirability ratings better than the alternatives we considered. The only different result in this case was that the detector-1 model appears to outperform the detector-2 model, possibly because the detector-2 model is more complex and thus prone to overfitting.

In Experiment 1C, we examined whether judgments on the rating task could be used to quantitatively predict choices for individual participants. To this end, we had each participant rate 400 items and then make binary choices between an additional 100 pairs of items. We fit the choice models to the rating data and then used Equation 11 to compute the choice probabilities for each pair. Figure 6 shows the choice prediction accuracy for each

This document is copyrighted by the American Psychological Association or one of its allied publishers.
This article is intended solely for the personal use of the individual user and is not to be disseminated broadly.



r	Counter-2
7	*red chili powder* + salsa
7	chili + black pepper
7	black pepper + vinegar + spicy tuna sushi
6	*red chili powder* + spicy tuna sushi
6	black pepper + salsa + olive oil
6	turkey + cayenne + black pepper

r	Tree
8	*red wine* + *red chili powder*
8	chili + orange zest + *red chili powder*
7	*red chili powder* + salsa
7	chili + olive oil + *red chili powder*
6	*red chili powder* + spicy tuna sushi
6	chili + coconut + *red chili powder*

Figure 4. Example of food combination ratings in Experiment 1A (top). Predictions and data for a single food combination (bottom). Nearest-neighbor combinations (as measured by the kernel function k), sorted by decreasing utility (r). Italics indicate ingredients that are shared with the target combination. See the online article for the color version of this figure.

model. The tree model and the counter-2 model are roughly equivalent in their predictive accuracy, but significantly better than the accuracies of the other models ($p < .05$ for all pairwise comparisons). Note that the absolute prediction accuracy is overall fairly low, but this is likely to reflect the limitation of our attribute space, which was constructed heuristically.

In summary, Experiments 1A to 1C provided quantitative support for the tree model using a realistic judgment task and complex food combinations. One weakness of these experiments is the use of uncontrolled stimuli; we do not really know what sort of attributes people use to represent ingredients, and this is a limiting factor in the predictive accuracy of our models. Thus, it is unclear whether the superior performance of the tree model relies on our idiosyncratic choice of attributes. Experiment 2 sought to complement Experiments 1A to 1C by using a strongly controlled, but necessarily more artificial, task. A further aim of Experiment 2 was to explore the learning processes underlying choice in combinatorial domains. In real life, people must learn about utilities from very sparse data, often just one or a few experiences. Experiment 2 was designed to test how different utility models compare with the ability of people to learn from sparse—but structured—data.

Table 2
Examples of Low- and High-Rated Combinations of Complex Ingredients in Experiment 1B

Low-rated	High-rated
garlic bread + mocha	pizza + grilled cheese
vinaigrette + banana sundae	chocolate milk + latte
tuna salad + banana sundae	ketchup + egg salad

Experiment 2

In Experiment 2, we again asked participants to make judgments about foods, but in this case, the foods were unfamiliar. Thus, participants could not appeal to domain knowledge and were forced to utilize the abstract food information provided to them. This allowed us to make a

This document is copyrighted by the American Psychological Association or one of its allied publishers.
This article is intended solely for the personal use of the individual user and is not to be disseminated broadly.

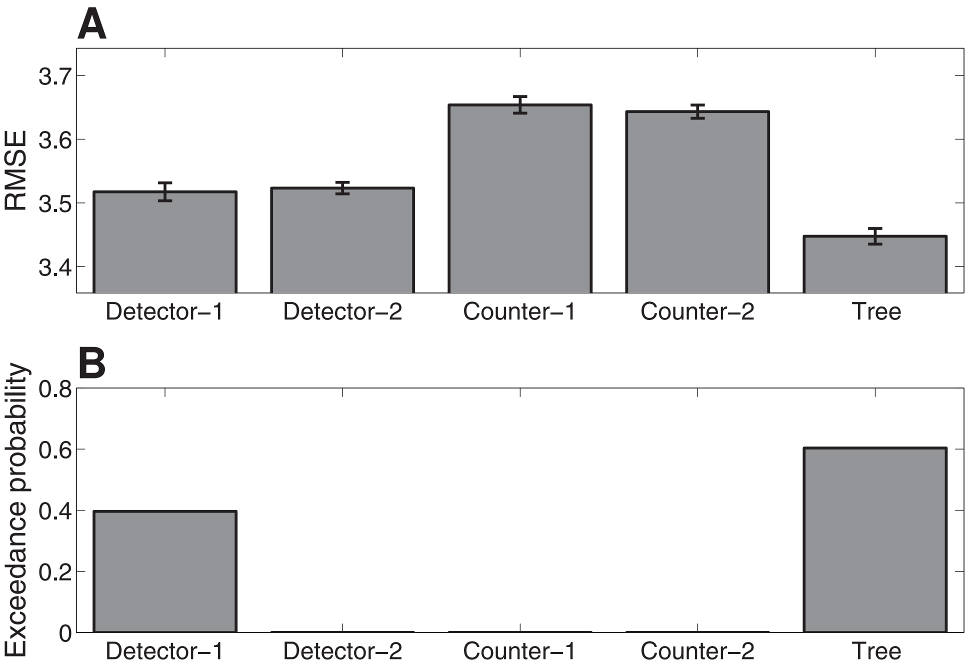


Figure 5. Model comparison: Experiment 1B. (A) Root mean-squared error (RMSE) for each model's predictions, computed on held-out data using a leave-one-out cross-validation procedure. (B) Exceedance probability for each model.

more strongly controlled comparison between the models. In particular, we presented participants with choice problems that were specifically designed to discriminate between the tree model and the alternative models. This experiment also provided an opportunity to gain insight into how participants learn about utilities from sparse, structured data.

Materials and Methods

Participants. Three groups of participants were recruited through Amazon Mechanical Turk for this experiment: 16 participants for the two calibration problems, 46 participants for Choice Problems 1 to 4, and 38 participants for Choice Problems 5 to 7.

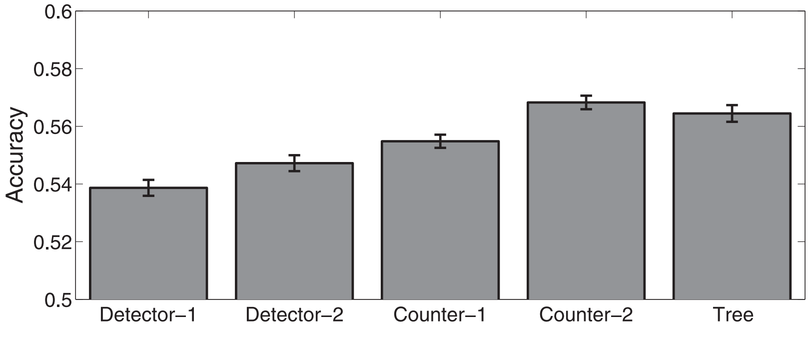


Figure 6. Model comparison: Experiment 1C. Choice prediction accuracy for each model trained on data from the rating task.

This document is copyrighted by the American Psychological Association or one of its allied publishers.
This article is intended solely for the personal use of the individual user and is not to be disseminated broadly.

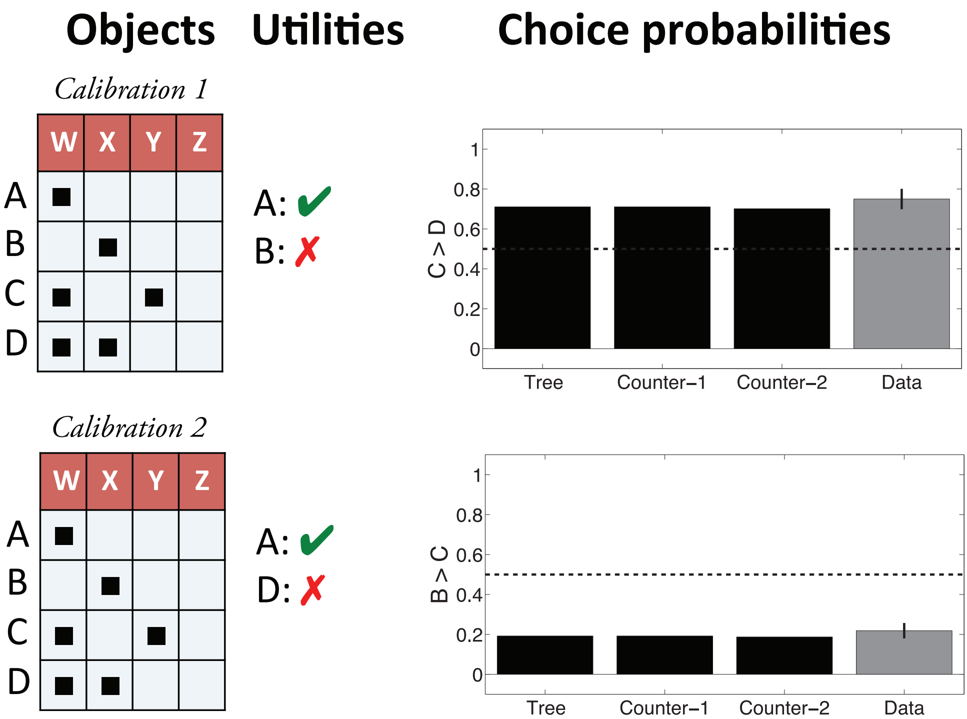


Figure 7. Results for calibration problems in Experiment 2. Each row corresponds to a separate choice problem. The left column shows the primitive attributes (W, X, Y, and Z) for a set of sandwiches (A, B, and C). The middle column shows the sampled sandwiches and their utilities (check mark [✓] indicates positive utility; [×] indicates negative utility). The right column shows the model and empirical choice probabilities; the *y*-axis label indicates the two options (e.g., $C > A + B$ means that larger *y* values denote a preference for option *C* over option $A + B$). The horizontal dashed line represents indifference between the two options. Error bars represent standard errors. See the online article for the color version of this figure.

Procedure. Each choice problem had the same format. Participants were told to imagine that they were visiting a foreign country, and for lunch each day they went to a different deli. All the ingredients and sandwich names were unfamiliar and labeled by made-up names; here, we label the sandwiches A, B, C, D, and the ingredients W, X, Y, and Z. Participants were shown the ingredients of each sandwich on the menu and then told that they sampled a few of these sandwiches or combos (half of one sandwich, half of another). For each sampled sandwich, a utility rating was indicated by check marks (positive) or "X" marks (negative), with the strength of positivity or negativity indicated by the number of marks. Finally, participants were given a choice between two novel sandwiches or combos. Participants rated their preference on a 5-point scale, which was then linearly rescaled to be a probability. Two of the choice problems were "calibration problems," used only to fit the parameters of the models (see Figure 7). The other seven choice problems (Figures 8 and 9) were selected so that the tree model would always (regardless of the parameter settings) make a qualitatively different prediction from the alternative models.

Model-fitting. Model predictions for choice probabilities were computed using Equation 11. In computing model predictions, we had to make somewhat arbitrary assumptions about the mapping from check marks and X marks to utilities; we assumed that each check mark corresponds to a utility of 10, and that each X mark corresponds to a utility of −10. Model predictions are fairly robust to other

This document is copyrighted by the American Psychological Association or one of its allied publishers.
This article is intended solely for the personal use of the individual user and is not to be disseminated broadly.

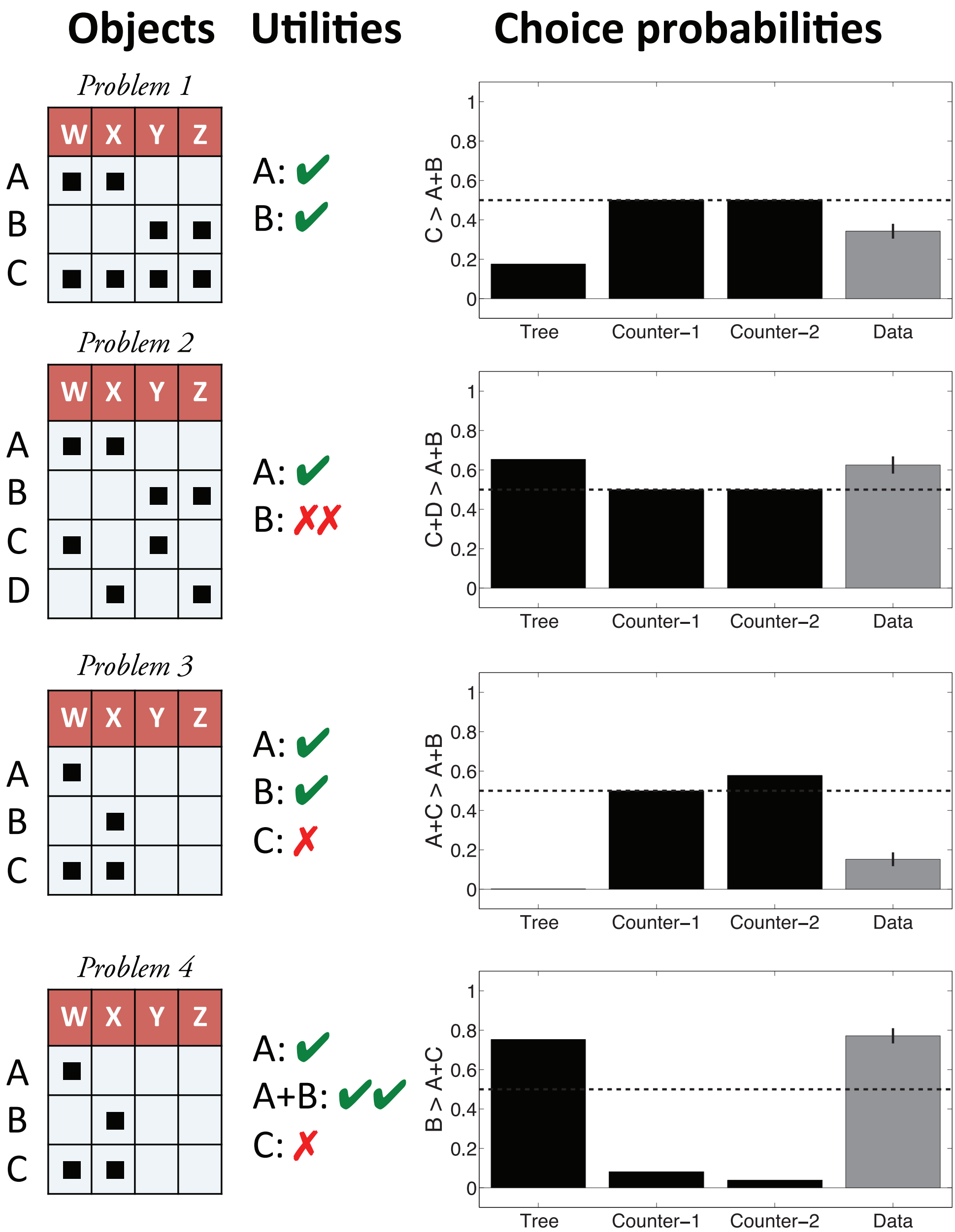


Figure 8. Results of Choice Problems 1 to 4 in Experiment 2. Each row corresponds to a separate choice problem. The left column shows the primitive attributes (W, X, Y and Z) for a set of sandwiches (A, B and C). The middle column shows the sampled sandwiches and their utilities (check mark indicates positive utility, × indicates negative utility). Multiple check marks [✓] or [×] marks indicate stronger (positive or negative, respectively) utilities. The right column shows the model and empirical choice probabilities; the Y-axis label indicates the two options (e.g., $C > A + B$ means that larger y values denote a preference for option C over option $A > B$). The horizontal dashed line represents indifference between the two options. Error bars represent standard errors. See the online article for the color version of this figure.

This document is copyrighted by the American Psychological Association or one of its allied publishers.
This article is intended solely for the personal use of the individual user and is not to be disseminated broadly.

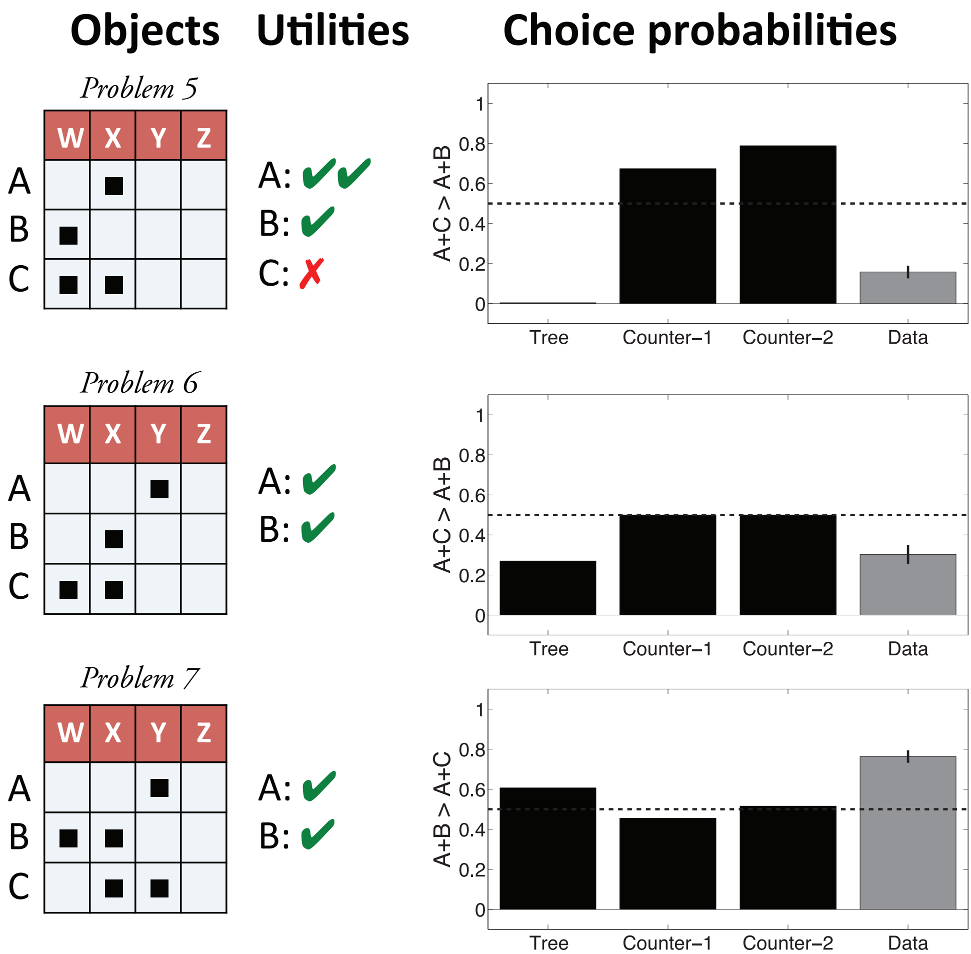


Figure 9. Results of Choice Problems 5 to 7 in Experiment 2. Each row corresponds to a separate choice problem. The left column shows the primitive attributes (W, X, Y and Z) for a set of sandwiches (A, B and C). The middle column shows the sampled sandwiches and their utilities (check mark indicates positive utility, × indicates negative utility). Multiple check marks [✓] or [×] marks indicate stronger (positive or negative, respectively) utilities. The right column shows the model and empirical choice probabilities; the y-axis label indicates the two options (e.g., $C > A + B$ means that larger y values denote a preference for option C over option $A > B$). The horizontal dashed line represents indifference between the two options. Error bars represent standard errors. See the online article for the color version of this figure.

This document is copyrighted by the American Psychological Association or one of its allied publishers.
This article is intended solely for the personal use of the individual user and is not to be disseminated broadly.

mappings, provided they are symmetric. Furthermore, the free parameter τ effectively induces a scale on the utilities, and thus by fitting τ to choice data from the calibration problems, we are removing some of the arbitrariness of the utility mapping.

Results and Discussion

The results of all the choice problems are shown in Figures 7 to 9. The calibration problems (see Figure 7) were used only to fit the τ parameter for each model; these problems are not, by themselves, of any theoretical interest, as they do not discriminate between the different models. Problems 1 to 7, shown in Figures 8 and 9, were chosen specifically to discriminate between the models (here we only consider the counter models, as the detector models performed poorly on the previous experimental data). For some of these problems (1, 2, and 6), both counter models are perfectly indifferent between the two options; for the other problems, the counter models showed a preference *opposite* to the preference shown by the tree model.

We found that for all problems, participants showed a significant deviation from indifference (see Table 3). This deviation was always consistent with the predictions of the tree model and in the opposite direction of the counter models' predictions (for those problems in which the counter models deviated from indifference, specifically Problems 3, 4, 5, and 7). Overall, the tree model produces predictions that are in quantitative agreement with the empirical choice probabilities ($r = .97$, $p < .0001$; see Figure 10). Excluding the two calibration problems, the correlation is still $r = .97$. In contrast, the counter models was nonsignificant ($r = -0.15$, $p = .71$ for counter-1; $r = -0.23$, $p = .55$ for counter-2). Thus, our results appear to invalidate the class of counter models.

To gain deeper insight into the nature of the preferences in our task, we now examine each choice problem in detail. In what follows, we will sometimes use A, B, and C to refer to objects, and sometimes to refer to object features; when this usage might be ambiguous, we explicitly state whether the symbol refers to an object or an object feature.

Choice Problem 1. Sandwich C has the same primitive attributes as the combination of Sandwiches A and B. Thus, the counter models will always be indifferent between C and A + B, regardless of the utilities of A and B. In contrast, the tree model adds an "object bonus," which favors A + B over C; specifically, features corresponding to Objects A and B earn credit in addition to the credit earned by their features, whereas C (being a novel object) lacks this extra credit.

Table 3
Results of t Tests Comparing Average Choice Probability (Shown in Figures 8 and 9) to Indifference (.5) for Each Choice Problem

Choice problem	*t* statistic	*p* value
1	$t(45) = 4.20$	$p < .01$
2	$t(45) = 2.85$	$p < .01$
3	$t(45) = 9.89$	$p < .001$
4	$t(45) = 7.02$	$p < .001$
5	$t(37) = 10.74$	$p < .001$
6	$t(37) = 4.09$	$p < .01$
7	$t(37) = 8.44$	$p < .001$

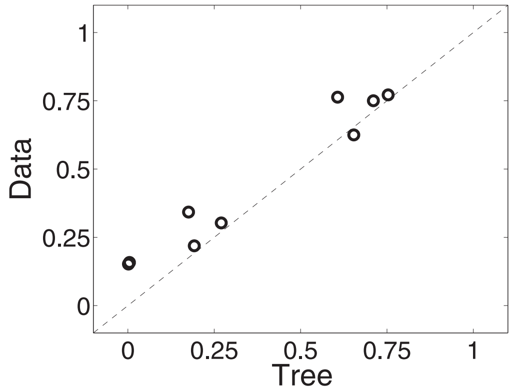


Figure 10. Relation between tree model predictions and empirical data for all nine problems in Experiment 2.

Choice Problem 2. A similar line of reasoning applies to this choice problem. Because C + D and A + B have the same features, the counter models are indifferent between the two options. In contrast, the tree model adds an "object penalty" to A + B (because B was associated with a large negative utility) which favors C + D.

Choice Problem 3. In this case, the counter-1 model is indifferent between Options A + C and A + B, because the positive credit assigned to features W and X (due to Objects A and B, respectively) is canceled out by the negative credit assigned to the same features by Object C (see Figure 11). The counter-2 model is not indifferent, because $u(A + C) = 2{*}w_W + w_X - w_{W*X} > 0$; however, the counter-2 model's slight preference for A + C is inconsistent with participants' (and the tree model's) preference for A + B. The tree model prefers A + B because Objects A and B were previously rewarded, and therefore receive an object bonus, whereas Object C receives an object penalty.

Choice Problem 4. In this case, both counter models strongly prefer A + C over B, whereas participants and the tree model prefer B. For both counter models, the W feature receives positive credit, but for the tree model, the W feature receives a small penalty (see Figure 12). This is because the object features are better predictors of utility and therefore "explain away" the attribute features. Conse-

This document is copyrighted by the American Psychological Association or one of its allied publishers.
This article is intended solely for the personal use of the individual user and is not to be disseminated broadly.

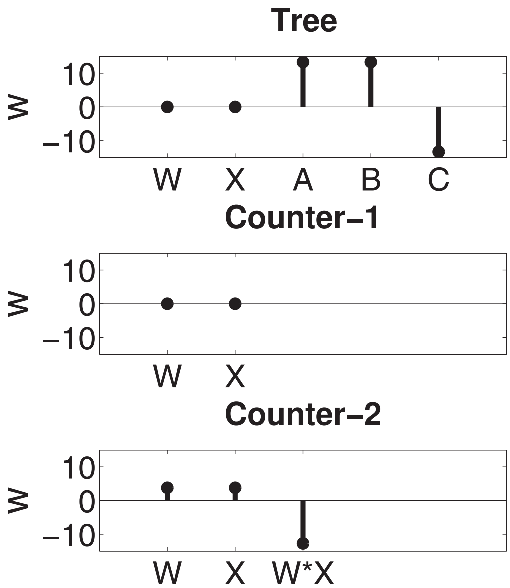


Figure 11. Inferred feature weights for Choice Problem 3. Each point on the *x*-axis corresponds to a feature, and the height of the stem indicates the posterior mean feature weight. Primitive attribute features are denoted by W and X; conjunctive features are denoted by an asterisk (*; e.g., W*X is the conjunction of W and X); object features are denoted by A, B, and C. Weights for three models are shown: the tree model (top), the first-order counter model (middle), and the second-order counter model (bottom).

quently, the preference for B is determined by the large positive weight accruing to Object Feature B, whereas the negative weight accruing to C cancels out the positive weight accruing to A when A and C are combined.

Choice Problem 5. A similar line of reasoning applies to this choice problem. For the tree model, the object features explain away the attribute features, such that Object Features A and B accrue large positive weights, and C accrues a large negative weight. This results in a preference for A + B over A + C, consistent with the empirical data. In contrast, the counter models both learn a positive weight for X (due to the large positive utility assigned to A, which is not canceled out by the smaller negative utility assigned to C), and this results in a preference for A + C over A + B.

Choice Problem 6. Here again, the counter models are indifferent between the two options because the weights on the attribute features are perfectly balanced between Options A + C and A + B. The tree model, however, receives an object bonus for Objects A and B, tilting the preference toward A + B.

Choice Problem 7. The counter models are nearly indifferent between Options A + B and A + C, because no attribute feature or conjunction of attribute features perfectly discriminates between the two combinations. The tree model shows a stronger preference for A + B because of an object bonus for the Objects A and B.

General Discussion

In this article, we developed a theory of compositional utility (the tree model) and tested the theory's predictions in a series of experiments. Experiments 1A and 1B showed that our theory could quantitatively predict judgments in a naturalistic food rating task, excelling several alternative models inspired by classical utility models. Experiment 2, using strongly controlled stimuli (unfamiliar foods), provided more decisive evidence in favor the tree model: For all choice problems, human preferences aligned

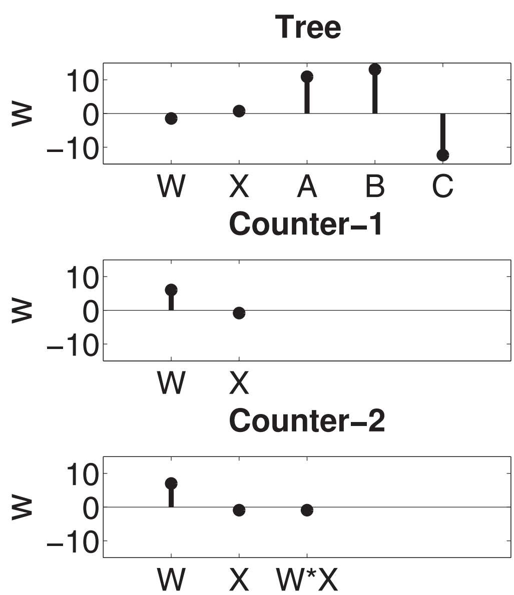


Figure 12. Inferred feature weights for Choice Problem 4. Each point on the *x*-axis corresponds to a feature, and the height of the stem indicates the posterior mean feature weight. Primitive attribute features are denoted by W and X; conjunctive features are denoted by '*'; object features are denoted by A, B, and C. Weights for three models are shown: the tree model (top), the first-order counter model (middle), and the second-order counter model (bottom).

This document is copyrighted by the American Psychological Association or one of its allied publishers.
This article is intended solely for the personal use of the individual user and is not to be disseminated broadly.

with the tree model and not the alternatives. In some cases, the discordance was dramatic, with the alternative models strongly predicting preferences in the opposite direction from participants' preferences.

The crucial contributing factor to the success of the tree model was an abstract representation of objects that goes beyond simple attributes or conjunctions of attributes. Consistent with this characteristic of the tree model, people appear to assign credit to objects in addition to low-level attributes. The composition laws of the tree model formalize the notion of an "object bonus" that privileges previously experienced objects (as opposed to attributes per se) in the utility calculus. In appealing to a privileged representation of objects, our theory joins a large swathe of research that asserts the pivotal role played by objects in many areas of cognition (Feldman, 2003; Kanwisher & Driver, 1992; Scholl, 2001; Spelke, 1990; Téglás et al., 2011).

Importantly, our notion of an object is compositional: Complex objects can be formed from simpler objects through set operations. This endows our model with productivity (an infinite number of objects can be represented in this way) and systematicity (the ability to represent an equivalence class of structurally similar objects). When combined with a probability distribution over utilities, this compositional object representation is able to capture the nuances of how object structure influences preferences (see also Barron, Dolan, & Behrens, 2013, for another interesting approach). We view our model as bridge between theories of utility developed in behavioral economics and recent work on the probabilistic language of thought (Goodman et al., 2008, 2011; Kemp et al., 2008; Piantadosi et al., 2012), which attempts to synthesize logical and probabilistic rules of cognition within a single unifying framework.

It might be objected that this complex machinery is unnecessary for understanding what superficially appears to be a familiar problem—learning nonlinear functions. For example, the case of broccoli ice cream could be construed as an "exclusive-or" problem, in which the utility function assigns positive utility to the individual components but negative utility to their combination (Minsky & Papert, 1969). This problem has been studied extensively in the psychology of associative learning, in which it is known as "negative patterning" (Whitlow & Wagner, 1972), and mathematical models have been developed to account for this learning (e.g., Pearce, 1994; Schmajuk & DiCarlo, 1992). More generally, the problem of associative learning with stimulus compounds has received extensive attention, and so one might reasonably wonder whether existing models can be brought to bear here. Our view is that such approaches cannot adequately address the challenges of utility representation in combinatorial domains because they typically rely on a fixed input representation (e.g., a neural network whose input units encode a finite number of stimuli). However, humans can evaluate arbitrarily complex combinations of stimuli. At the same time, our experimental results show that this flexibility is balanced by the constraints of the utility tree, which reflect assumptions about the nature of object representations. We believe this combination of flexibility and object-based constraints is a hallmark of human decision making.

One limitation of our modeling in this article is that our representations of food combinations lack the right kinds of abstractions. Broccoli ice cream is represented as a composite of broccoli and ice cream, but humans are also likely to represent this as a kind of "vegetable dessert." The key problem is that the tree structure is defined entirely in terms of sets, whereas a more cognitively plausible representation would also include high-level annotations. We can remedy this problem by adding internal nodes to the tree that represent these annotations. Choosing the appropriate annotations is tricky, and we have therefore decided to leave this task to future work. Another approach to this problem would be to add conjunctive features to the terminals of the tree structure, as a way of capturing some of the high-level annotations that ground out in particular conjunctions (e.g., "vegetable dessert" might be approximated by the conjunction of the attributes "vegetable" and "sweet"). We implemented a version of this second-order tree model and found that it did not improve predictive accuracy over the first-order tree model. In the future, we plan to design experiments that specifically target the issue of abstraction, which will allow us to ascertain whether such representations are im-

This document is copyrighted by the American Psychological Association or one of its allied publishers.
This article is intended solely for the personal use of the individual user and is not to be disseminated broadly.

portant for capturing human judgments (as we intuitively feel they should be).

Another limitation of our modeling is that we have attempted to directly measure a cardinal utility representation, despite the conviction of most decision theorists that preferences can only be measured ordinally (Pareto, 1906). Although we empirically demonstrated the predictive power of our models in capturing people's quantitative utility judgments, it would also be useful to test them using ordinal measures and other measurement notions as developed in revealed preference theory.

To conclude, we highlight two interesting directions for future research. First, we have assumed a "batch" setup, in which all the experienced utilities are fed into the model and the Bayesian inference engine spits out utility predictions. Alternatively, we could consider the "online" setup of reinforcement learning (Sutton & Barto, 1998), in which the utility predictions must be learned incrementally. This could be studied experimentally in the multiarmed bandit setting, in which an agent is presented with a number of options (arms) that stochastically generate reward when selected. Our tree model could be applied in the case in which the arms are represented hierarchically. A second interesting direction for future research pertains to the setting in which the tree structure of objects is unknown. In this case, the agent is faced with the problem of inferring both the tree structure and the utility function. One approach to this problem would be to use structure learning algorithms (Gershman, Norman, & Niv, 2015; Kemp & Tenenbaum, 2008) to discover the underlying tree. It is an open question how this structure learning process might interact with reasoning about utility functions.

References

Bacchus, F., & Grove, A. (1995). Graphical models for preference and utility. In *Proceedings of the eleventh conference on uncertainty in artificial intelligence* (pp. 3–10). Montreal, Quebec, Canada: Morgan Kaufmann.

Barron, H. C., Dolan, R. J., & Behrens, T. E. (2013). Online evaluation of novel choices by simultaneous representation of multiple memories. *Nature Neuroscience, 16,* 1492–1498.

Bentham, J. (1879). *An introduction to the principles of morals and legislation*. Broadbridge Mill, UK: Clarendon Press.

Braziunas, D., & Boutilier, C. (2009). Elicitation of factored utilities. *AI Magazine, 29,* 79.

Carroll, J. D., & De Soete, G. (1991). Toward a new paradigm for the study of multiattribute choice behavior: Spatial and discrete modeling of pairwise preferences. *American Psychologist, 46,* 342–351.

Feldman, J. (2003). What is a visual object? *Trends in Cognitive Sciences, 7,* 252–256.

Fishburn, P. C. (1967). Interdependence and additivity in multivariate, unidimensional expected utility theory. *International Economic Review, 8,* 335–342.

Fodor, J. A. (1975). *The language of thought*. Cambridge, MA: Harvard University Press.

Gershman, S. J., Norman, K. A., & Niv, Y. (2015). Discovering latent causes in reinforcement learning. *Current Opinion in Behavioral Sciences, 5,* 43–50.

Gilboa, I., & Schmeidler, D. (1995). Case-based decision theory. *The Quarterly Journal of Economics, 110,* 605–639.

Gonzalez, C., Lerch, J. F., & Lebiere, C. (2003). Instance-based learning in dynamic decision making. *Cognitive Science, 27,* 591–635.

Goodman, N. D., Tenenbaum, J. B., Feldman, J., & Griffiths, T. L. (2008). A rational analysis of rule-based concept learning. *Cognitive Science, 32,* 108–154.

Goodman, N. D., Ullman, T. D., & Tenenbaum, J. B. (2011). Learning a theory of causality. *Psychological Review, 118,* 110–119.

Gorman, W. M. (1968). The structure of utility functions. *The Review of Economic Studies, 35,* 367–390.

Hertwig, R., & Erev, I. (2009). The description–experience gap in risky choice. *Trends in Cognitive Sciences, 13,* 517–523.

Kahneman, D., & Snell, J. (1992). Predicting a changing taste: Do people know what they will like? *Journal of Behavioral Decision Making, 5,* 187–200.

Kahneman, D., Wakker, P. P., & Sarin, R. (1997). Back to Bentham? Explorations of experienced utility. *The Quarterly Journal of Economics, 112,* 375–406.

Kanwisher, N., & Driver, J. (1992). Objects, attributes, and visual attention: Which, what, and where. *Current Directions in Psychological Science, 1,* 26–31.

Keeney, R. L., & Raiffa, H. (1976). *Decisions with multiple objectives: Preferences and value trade-offs*. Malden, MA: Wiley.

Kemp, C., Goodman, N. D., & Tenenbaum, J. B. (2008). Theory acquisition and the language of thought. In *Proceedings of thirtieth annual meeting of the cognitive science society* (pp. 1606–1611). Austin, TX: Cognitive Science Society.

This document is copyrighted by the American Psychological Association or one of its allied publishers.
This article is intended solely for the personal use of the individual user and is not to be disseminated broadly.

Kemp, C., & Tenenbaum, J. B. (2008). The discovery of structural form. *PNAS Proceedings of the National Academy of Sciences of the United States of America, 105,* 10687–10692.

Krantz, D., Luce, D., Suppes, P., & Tversky, A. (1971). *Foundations of measurement, Vol. I: Additive and polynomial representations*. New York, NY: Academic Press.

Lucas, C. G., Griffiths, T. L., Williams, J. J., & Kalish, M. L. (2015). A rational model of function learning. *Psychonomic Bulletin & Review, 22,* 1–23.

McFadden, D. (1973). Conditional logit analysis of qualitative choice behavior. In P. Zarembka (Eds.), *Frontiers of Econometrics*. New York, NY: Academic Press.

Minsky, M. L., & Papert, S. A. (1969). *Perceptrons*. Cambridge, MA: MIT Press.

Moschitti, A. (2006). Making tree kernels practical for natural language learning. In *Proceedings of the eleventh international conference on European Association for Computational Linguistics* (Vol. 6, pp. 113–120). Trento, Italy.

Pareto, V. (1906). *Manuale di Economia Politica* [Manual of political economy]. New York, NY: A. M. Kelley (translated from French edition of 1927).

Pearce, J. M. (1994). Similarity and discrimination: A selective review and a connectionist model. *Psychological Review, 101,* 587–607.

Piantadosi, S. T., Tenenbaum, J. B., & Goodman, N. D. (2012). Bootstrapping in a language of thought: A formal model of numerical concept learning. *Cognition, 123,* 199–217.

Rasmussen, C., & Williams, C. (2006). *Gaussian processes for machine learning*. Cambridge, MA: MIT Press.

Rescorla, R. A., & Wagner, A. R. (1972). A theory of Pavlovian conditioning: Variations in the effectiveness of reinforcement and nonreinforcement. In A. Black & W. Prokasy (Eds.), *Classical conditioning II: Current research and theory* (pp. 64–99). New York, NY: Appleton-Century-Crofts.

Schmajuk, N. A., & DiCarlo, J. J. (1992). Stimulus configuration, classical conditioning, and hippocampal function. *Psychological Review, 99,* 268–305.

Scholl, B. J. (2001). Objects and attention: The state of the art. *Cognition, 80,* 1–46.

Spelke, E. S. (1990). Principles of object perception. *Cognitive Science, 14,* 29–56.

Stephan, K. E., Penny, W. D., Daunizeau, J., Moran, R. J., & Friston, K. J. (2009). Bayesian model selection for group studies. *NeuroImage, 46,* 1004–1017.

Stewart, N., Chater, N., & Brown, G. D. (2006). Decision by sampling. *Cognitive Psychology, 53,* 1–26.

Strotz, R. H. (1957). The empirical implications of a utility tree. *Econometrica: Journal of the Econometric Society, 25,* 269–280.

Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction*. Cambridge, MA: MIT Press.

Téglás, E., Vul, E., Girotto, V., Gonzalez, M., Tenenbaum, J. B., & Bonatti, L. L. (2011). Pure reasoning in 12-month-old infants as probabilistic inference. *Science, 332,* 1054–1059.

Tversky, A. (1972). Elimination by aspects: A theory of choice. *Psychological Review, 79,* 281–299.

Tversky, A., & Sattath, S. (1979). Preference trees. *Psychological Review, 86,* 542–573.

Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-sne. *Journal of Machine Learning Research, 9,* 2579–2605.

Vishwanathan, S. V. N., & Smola, A. J. (2002). Fast kernels on strings and trees. In *Proceedings of Neural Information Processing Systems* (pp. 569–576). Vancouver, British Columbia, Canada.

Von Neumann, J., & Morgenstern, O. (1947). *Theory of games and economic behavior*. Princeton, NJ: Princeton University Press.

Wakker, P. P. (1989). *Additive representations of preferences: A new foundation of decision analysis*. Dordrecht, the Netherlands: Kluwer Academic.

Whitlow, J. W., Jr., & Wagner, A. R. (1972). Negative patterning in classical conditioning: Summation of response tendencies to isolable and configural components. *Psychonomic Science, 27,* 299–301.

Wilson, T. D., & Gilbert, D. T. (2005). Affective forecasting knowing what to want. *Current Directions in Psychological Science, 14,* 131–134.

Received May 31, 2015
Revision received January 9, 2016
Accepted January 25, 2016 ■

This document is copyrighted by the American Psychological Association or one of its allied publishers.
This article is intended solely for the personal use of the individual user and is not to be disseminated broadly.