Rational representations of uncertainty: a pluralistic approach to bounded rationality

Abstract

An increasingly prevalent approach to studying human cognition is to construe the mind as optimally allocating limited cognitive resources among cognitive processes. Under this *bounded rationality* approach (Icard 2018, Simon 1980), it is common to assume that resource-bounded cognitive agents approximate rational solutions to inference problems, and that much of the bias and variability in human performance can be explained in terms of the approximation strategies we employ. In this paper, we argue that this inference chain is too quick: boundedness does not always imply approximation, and in fact approximation can (in certain cases) be more costly than exact inference. We advocate for a pluralistic approach to bounded rationality, which optimizes over different means of representing uncertainty, in addition to different methods of manipulating those representations. We outline a framework for this optimality analysis, and present a case study demonstrating how approximate solutions can be costlier than exact solutions, thus illustrating that boundedness and optimal performance are not always in tension.

1 Introduction

1.1 Background and motivation

Much of our everyday cognitive and perceptual activity involves inference under uncertainty: across a wide range of contexts, we must make judgments that are undetermined by the (often sparse and noisy) data available to us. *Are these pants black or dark blue? Will my friend enjoy this horror movie, or will it make them uncomfortable? Is this car merging into my lane, or did they leave their blinker on by mistake?* In the study of human cognition, it has become increasingly common to interpret our cognitive capacities through this lens, an approach known as *rational analysis* (Anderson 1990). The rational analysis methodology is motivated by an assumption that the human mind has adapted to solve certain kinds of environmentally-grounded decision problems with limited information, and we can gain key insights into human cognition by precisely characterizing these problems and their optimal solutions.

Rational analysis is traditionally formulated at the computational level of analysis (Marr 1982), aiming to capture the formal structure of the inference problems we solve (i.e.: the information content of inputs and outputs), and comparing human performance against the optimal solutions to those problems. As such, rationalist models are typically posited as useful *descriptions* of human behavior, rather than genuine *explanations* of the neural and cognitive mechanisms underlying that behavior. More recent work, however, has sought to bridge this explanatory gap, extending the methodology of rational analysis to the algorithmic level of description. This approach, referred to as *boundedly rational analysis* (Icard 2018) or *resource-rational analysis* (Lieder & Griffiths 2020), explicitly accounts for the limited computational resources (e.g.: time, memory, metabolic energy) with which the mind operates. Rather than modeling cognitive

activity as (approximately) optimal inference under uncertainty, boundedly rational analysis models cognitive activity as the (approximately) optimal allocation of cognitive resources. Recent work has leveraged this assumption to show how many of the apparent biases and errors that characterize human reasoning (Tversky & Kahneman 1974) actually reflect optimal performance under certain assumptions about the cost of computation (e.g.: Lieder et al 2012, Vul et al 2014).

1.2 Our contribution

In this paper, we argue that a certain common approach to boundedly rational cognitive modeling limits itself to an unnecessarily narrow scope of plausible cognitive models, which risks glossing over certain details that are potentially critical to the analysis. This approach is characterized by first defining a particular computational-level representation of a problem, deriving the optimal solution to that problem for an unbounded agent, then considering the optimal algorithm through which an agent with finite computational resources should approximate that solution. Underlying this approach are certain assumptions about how the agent can represent uncertainty and manipulate those representations: at a computational level, the optimal solution involves exact computation over explicit representations of uncertainty (e.g.: Bayesian posterior inference over a prior probability distribution- Griffiths et al 2008). At the algorithmic level, the most common assumption in the literature is that agents leverage sampling-based algorithms for approximating probabilistic computations (e.g.: Bonawitz et al 2014, Denison et al 2014).

We argue that the focus on approximating optimal solutions, and the particular focus on sampling-based approximations, is neither immediately demanded nor immediately justified by the assumptions of boundedly rational analysis. First, approximating the optimal solution is not always the most rational response for an agent with limited

computational resources; in fact, there are many cases in which approximation may be *less rational* than exact computation. More generally, we argue that this approach glosses over another potentially relevant dimension of optimization: the representations themselves. Given that there exist plausible neurophysiological accounts to support a range possible representations [cite], as well as preliminary behavioral evidence suggesting some flexibility in how we represent uncertainty [cite], we advocate for a more pluralistic approach to bounded rationality which optimizes over representational forms *and* algorithms for manipulating those representations. That is, rather that starting with a fixed representation of a problem, and considering how to manipulate that representation in a resource-rational way, we should consider the possibility that resource-bounded cognitive agents can flexibly adjust how they represent uncertainty in the first place, allowing them to learn representations which enable efficient manipulation. When we consider this possibility, we find that there are important bidirectional interactions between how we represent uncertainty for a given task, and how we can efficiently manipulate that representation to solve the task.

1.3 Outline

In the next section we provide more detail on the motivation and use of rational analysis, concerns about the explanatory capacity of rationalist cognitive models, and how boundedly rational analysis seeks to resolve these concerns. We then review recent work on boundedly rational cognitive modeling, how the scope of this work may be overly constrained, and what this implies about the rationalist justification of such models. Finally, we consider what the scope of our focus *ought* to be, and the requirements for an analysis framework that covers the appropriate scope.

In section three, we sketch out the requirements for such a framework, and point to some existing formal tools that are well-suited for the task. In particular, we show how

Probabilistic Programming Languages (PPLs) provide a unified framework for formalizing both a space of representations, and a space of algorithms for manipulating those representations, in a way that exposes certain trade-offs relevant for our analysis. Further, we argue that parameterized complexity theory (Blokpoel et al 2010, Downey & Fellows 2012) provides a useful lens through which to characterize these trade-offs in a way that supports joint optimization. We then provide a simple case study to demonstrate that, even with a fairly restricted problem space, this optimization can be quite non-trivial. As we shall argue, however, a fully formalized framework for this joint optimization requires attention to certain trade-offs and dimensions that are under-explored in the current literature. In section 4, we point to three such dimensions, describe how they can be relevant to the analysis, and motivate a way to unify these trade-offs into a single framework for optimization. Finally, we conclude in section 5 with a brief summary of our findings, before considering future directions for this research.

2 Background

While not an entirely novel concept (Simon 1955, 1980), bounded rationality has seen a recent surge of interest in cognitive science and psychology, largely motivated by an apparent tension between two different bodies of psychological research. Here we provide more background on these two approaches to studying the mind, how bounded rationality seeks to resolve the tension between them, and whether current approaches can fulfill this purpose.

2.1 Rational analysis

The study of human cognition faces a persistent identifiability problem. As we cannot directly observe or intervene on a subject's cognitive states, we generally have to rely on

(often sparse and noisy) behavioral data to distinguish hypotheses. Furthermore, the space of hypotheses (i.e.: high-level cognitive models) is largely unbounded in the absence of any strong theoretical assumptions. Given the sparsity of available data streams, relative to the vast space of possible hypotheses, there will usually be many (sometimes infinitely many) competing explanations compatible with the same data (Pylyshyn 1980). Rational analysis seeks to address this problem by narrowing our focus: by considering models that provide "optimal" solutions to the problems being solved by the mind, we can both reduce the space of possible alternatives to consider, and provide more quantifiable metrics for comparing competing models (Anderson 1990). Thus, a rational analysis of cognitive behavior proceeds by identifying the problems solved by the mind, developing normative models of the ideal solutions to those problems, and comparing human performance against those ideal solutions. Importantly, this approach is typically framed at the computational level of analysis (Marr 1982), aiming to characterize our cognitive behavior in terms of rational inferences while remaining agnostic about the cognitive or neural mechanisms underlying these inferences.

While there are multiple formalizations of rational analysis, the most prevalent by far is the Bayesian implementation (Chater & Oaksford 2007, Griffiths, Kemp, & Tenenbaum 2008). While our arguments are aimed at rational analysis more generally, grounding these arguments in a particular implementation will help illustrate them more saliently, and we choose the Bayesian implementation due to its tremendous presence in modern cognitive science. Formally, we represent an agent's uncertainty over possible states of the world as a prior probability distribution $P(w)$. Given some evidence E, a Bayesian agent will update the degree to which they believe in world-state w according to Bayes' rule:

$$P(w|E) = \frac{P(E|w)P(w)}{P(E)} \tag{2.1}$$

where $P(E|w)$ denotes the probability of observing E given that w is the true state of

the world (i.e.: the *likelihood* of E), $P(w|E)$ is the observer's updated degree of belief in w (i.e.: the *posterior probability* of w), $P(w)$ is the observer's prior degree of belief in w (before observing any evidence), and $P(E)$ is the overall likelihood of observing evidence E. Faced with the problem of inferring the true hypothesis after observing evidence E, the most rational solution[1] is to compute equation (2.1) for each possible world state, and return the state w^* which maximizes the posterior probability $P(w^*|E)$ (de Finetti 1937, Huttegger 2013). This strategy provides a normatively ideal solution for inference under uncertainty and has therefore been widely used as a basis for rational analysis of human cognition.

The main concern when deriving a Bayesian model of some cognitive capacity is how the agent represents the probability distributions in equation 2.1. The most common approach is to assume that the agent has some internal generative model, which specifies a set of variables (both observable and latent), and a set of probabilistic causal relations between these variables. To make this more concrete, we introduce a simple demonstration of such a model, which we will refer to throughout the rest of the paper as an illustrative example. To this end, imagine we are watching an agent navigate some environment (e.g.: a shopping mall). We observe the agent's first few steps $x_1, \ldots, x_{t-1}$, and wish to predict the agent's next step x_t (for example, to catch up with them and return something they have dropped). Figure 1a depicts such a task.

[1] In the sense that no other strategy can outperform this strategy

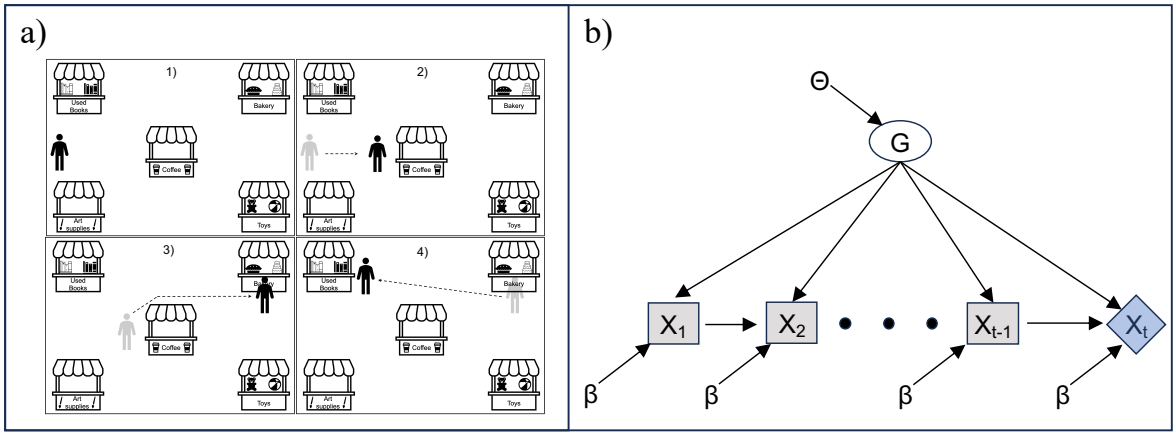

Figure 1: Illustration of an action-prediction task (panel a) and a simplified "rational planning" model for solving such a task (panel b). Panel a) depicts an agent navigating a shopping mall, where each sub-panel depicts the agent's next few steps. The observer's goal is to predict the agent's next steps. Panel b) depicts a simplified "rational planning model" (Baker et al 2009) for solving such a task. Variables in grey boxes are observed (i.e.: the agent's previous steps). Variables in circles are posited latents (i.e.: the agent's goal state). Variables shaded in blue are the targets of inference (i.e.: the agent's next step). Variables without borders are parameters that, together with the structure of the model, define a joint probability distribution over all variables in the model.

Behavioral inference tasks like these have been widely studied through the Bayesian framework. Figure 1b depicts a simplified version of a "rational planning model," a common generative model used to study this capacity (Baker et al 2009). Under a rational planning model, we assume that the agent has some latent goal state G (e.g.: to acquire a certain type of item), where the parameter Θ captures the prior probability that the agent will have a particular goal (i.e.: the agent's preferences over different states of the world). At each step, we assume that the agent will behave "rationally" up to some degree of error parameterized by β. This means that the agent will, with probability $1 - \beta$, move in the direction of a shortest path from their current location to

a goal location, or will move in a random direction with probability β. This assumption provides the likelihood term $P(x_t|G, x_1, \ldots, x_{t-1}; \beta)$ (i.e.: the probability of taking a particular action, given the agent's goal and prior actions), and the parameter Θ provides the prior distribution over goals $P(G; \Theta)$. Thus, this model encodes all of the information necessary to compute the posterior distribution in equation 2.1. This general approach- encoding a probability distribution in a generative model, and manipulating that model via Bayesian inference- has been used to model nearly every aspect of human cognition, including object perception and categorization (Kersten et al 2004, Salakhutdinov et al 2012), language production and interpretation (Goodman & Frank 2016), a range of intuitive theories such as physics (Smith & Vul 2013) and psychology (Baker et al 2011, Jara-Ettinger et al 2016), and the very process of cognitive development itself (Gopnik & Wellman 2012).

2.2 How rational are we, really?

Despite the success of rational analysis at accurately *describing* human inferences across a wide range of domains, there remain concerns (both theoretical and empirical) about the viability of rationalist models (and Bayesian models in particular) as *explanations* of human cognition. The primary theoretical concern is tractability: outside of simple cases, the computations underlying rational statistical analysis are generally intractable, in the sense that the amount of computation required increases exponentially (or worse) in the size of the input. In causal inference, for example, the number of possible causal structures over a set of variables increases exponentially in the number of variables. In order to perform exact posterior inference, the observer would have to compute the posterior probability (equation 2.1) of each possible structure, then return the structure with the highest posterior probability. Furthermore, many Bayesian cognitive models involve either continuous or infinitely recursive hypothesis spaces (e.g.: Griffiths &

Ghahramani 2005), rendering exact inference completely infeasible. Thus, a rationalist explanation of human cognition must address how we, as cognitive agents perform these seemingly intractable computations at all, much less quickly enough to make real-time decision (Jones & Love 2011).

On the empirical side, there are many cases in which the claims of rationality underlying this framework don't seem to manifest in human responses. In fact, it is quite well established that human statistical judgments contain systematic errors and biases (Tversky & Kahneman 1974) that deviate from the predictions of rational Bayesian inference. For example, our estimations are often improperly biased or "anchored" towards numerical values we have previously considered, even when those values have no relation to the values we are estimating (Epley & Gilovich 2006); we consistently over-weigh the probability of unlikely events with extreme consequences (Lichtenstein et al 1978). Furthermore, there is often a great deal of variability in human responses, both between and within individuals (Mozer et al 2008), which does not match the behavior predicted by a rational Bayesian decision-making. In particular, a rational Bayesian agent should always "posterior maximize," i.e.: deterministically choose the hypothesis with the highest posterior probability. In many cognitive studies, however, there is significant variability among participants' responses, and the overall empirical distribution of these responses tends to match the Bayesian posterior distribution, a phenomenon known as "posterior matching." While this may intuitively seem like an approximately rational strategy, it has been shown that posterior matching has (under a computation-level rational analysis) no rational justification, should therefore *not* be interpreted as evidence that people are (approximately) rational (Eberhardt & Danks 2011).

2.3 A different kind of rationality

The bounded rationality program seeks to address both the theoretical and empirical challenges to rational analysis with a single conceptual reframing. Whereas computation-level rational analysis models agents with unbounded cognitive resources, but limited information access, boundedly rational analysis explicitly considers the limited resources (e.g.: time, memory, etc.) to which the human mind has access. This new set of constraints introduces a different set of trade-offs: in general, more accurate solutions require more computation, which in turn makes them more costly. Thus, a boundedly-rational agent should weigh the benefit of having a more accurate solution against the increased cost of computing a more accurate solution, and allocate cognitive resources up to the point where the benefit of increased accuracy is outweighed by the cost.

This reframing has two benefits with respect to the aforementioned concerns. On the theoretical side, it helps alleviate concerns about intractability by suggesting that people are not actually performing optimal statistical inference (which in most cases would require a prohibitive amount of computation to implement), but are instead approximating these computations in a more efficient way. Second, many of these approximation methods involve random sampling, often in a fashion that produces biased or auto-correlated outputs (e.g.: MCMC sampling). Thus, boundedly rational analysis aims to show that our apparently sub-rational biases and errors actually reflect a rational allocation of limited cognitive resources, by using stochastic and potentially biased algorithms for approximating Bayesian inference.

Indeed, many apparent biases in human inferences have been shown to reflect the behavior of certain kinds of algorithms for approximating Bayesian inference: our tendency to anchor estimations to previously considered values, or to base our decisions on a small number of guesses, reflects the optimal behavior of certain kinds of sampling

algorithms when generating additional samples is costly (Bonawitz et al 2014, Lieder et al 2012, Vul et al 2014). The over-weighting of unlikely events with extreme consequences reflects optimal sampling behavior for certain resource-constrained algorithms that approximate Bayesian inference (Lieder et al 2018). Posterior-*matching* can be more rational than posterior *maximizing* under certain constraints on memory (Icard 2019). Thus, the bounded rationality paradigm seems to provide a promising resolution to both the theoretical and empirical concerns levied against the rational analysis framework.

2.4 The scope of boundedly rational cognitive models

The concerns and insights that motivated the bounded rationality paradigm suggest a certain intuitive approach to deriving boundedly-rational cognitive models. First, we define a problem (e.g.: inferring an agent's mental states from their behavior) at the computational-level, and compute the rational solution to that problem for an unbounded observer; in the Bayesian framework, this means defining a generative model of some relevant part of the world (e.g.: an agent's mental states and how those states causally relate to the agent's behavior), and using this model to compute a posterior distribution over possible answers. However, as discussed in the previous section, these computations are typically intractable for a resource-bounded agent. Thus, an optimal *bounded* agent should approximate this posterior inference as well as is rational[2], given their cognitive resources. While intuitively appealing, this approach raises several concerns.

The first concern is that approximation does not always make an intractable problem tractable: for many commonly occurring statistical inference problems, even approximate solutions (for any fixed degree of accuracy) cannot be tractably computed

[2]i.e.: up to the point where the cost of additional computation exceeds the benefit of additional accuracy

in general (Kwisthout et al 2011). Even when approximation does enable a tractable solution, it is not necessarily the case that approximating the ideal Bayesian solution yields the boundedly optimal solution. For example, there are cases in which non-Bayesian heuristics outperform approximate Bayesian inference with the same limited resources (Icard 2018). Thus, even if approximate Bayesian inference is tractable, there is no guarantee that such an algorithm is actually resource-rational without assuming substantial restrictions on the space of plausible algorithms, and the space of representations over which those algorithms operate.

This leads to a second, more general concern about these models: how do we determine the appropriate set of cognitive constraints, including how the agent represents uncertainty for a given problem, and the set of algorithms through which the agent can manipulate those representations? If our assumptions are too general or minimal, we risk glosses over important factors that can influence the true "cost" of a solution. For example, one common approach is to assert that the agent has a method for drawing unbiased, independent samples from the relevant posterior distribution at a fixed cost per sample, while remaining agnostic about the details of the sampling process itself (e.g.: Bonawitz et al 2014, Vul et al 2014). However, it is often the case that generating unbiased, independent samples from a posterior distribution requires just as much computation as exact posterior inference (or else requires a prohibitive number of random decisions to approximate). Thus, glossing over the details of the sampling process in this fashion makes it difficult to assess how well this approach can inform a resource-rational understanding of human cognition.

On the other hand, if our assumptions are too narrow, we risk omitting other possible representations or algorithms that may be more efficient. For example, another common approach is to assume a fixed approximation algorithm (e.g.: some form of MCMC sampling), and evaluate the optimal use of that algorithm (e.g.: the optimal number of

samples to draw) (e.g.: Dasgupta et al 2017, Lieder et al 2012, Milli et al 2021). While this exposes the relevant details of the sampling process, these details are only applicable if humans do, in fact, use algorithms with the same properties as those assumed in the model. This assumption is complicated, however, by the fact that there are often many approximation methods than can be implemented with the same core machinery posited by these models. That is, given the cognitive machinery required to implement, say, a Metropolis-Hastings algorithm (a form of MCMC algorithm [cite]), one could implement a range of alternate algorithms for approximating the same distribution, including various forms of exact inference, rejection sampling, particle filters, and other MCMC algorithms. This fact makes it difficult to assert that one particular approximation algorithm is *the* right one to use in an algorithmic-level cognitive model.

In response to these concerns, some have proposed a different approach to understanding how resource-bounded agents could possibly perform these seemingly intractable computations. In particular, this approach suggests that, rather than finding efficient algorithms for approximating intractable computations, perhaps the mind simply forms representations for which tractable solutions already exist (Kwisthout et al, 2011). This approach is motivated by insights from Fixed-Parameter Complexity theory (Downey & Fellows 2012), which aims to break down the computational cost of solving a problem into different "dimensions" that characterize the structure of the problem. That is, suppose we have a class M of intractable problems, and we identify a set of parameters of interest $K = \{k_1, \ldots, k_n\}$ which characterize individual instances of problems within this class (e.g.: the number of latent variables in the problem, the number of values that each variable can assume, etc.). Given these parameters of interest, the aim of parameterized complexity analysis is to determine whether it is possible to solve problems in M efficiently when the values of these parameters are held fixed, even as the size of the input increases arbitrarily. If this is the case, then the

parameters in K are said to be the *source of intractability* for M, and M is said to be *fp-tractable for* K. It has been shown, for example, that a common class of Bayesian inference problem which is generally intractable is fp-tractable for two particular parameters- the maximum number of latent variables in the network, and the degree of certainty of the most probable configuration of latent states (Blokpoel et al 2011).

2.5 Moving forward

The previous section suggests two distinct conceptual approaches to boundedly-rational cognitive modeling. In the first approach, we start with a computation-level representation of a problem (i.e.: a particular generative model), and consider the resource-optimal algorithm for manipulating that representation (either exactly or approximately). In the second approach, we start with a description of a *task*, and consider how we can construct a representation of the uncertainty in that task for which tractable solutions exist. This distinction between positing a "task" and positing a "representation" is a subtle but important one. Consider, for example, the task illustrated in Figure 1a. If we characterize the task as one of "goal inference," as is common in the Bayesian Theory of Mind literature, this entails certain assumptions and restrictions on how an agent represents the task (i.e.: it assumes that the agent explicitly represents a latent goal state for the agent). On the other hand, if we simply characterize the task in terms of the relevant inputs (the agent's environment and previous behavior) and outputs (the agent's next action), this imposes fewer assumptions on the agent's representation, and leaves more flexibility for the agent to "optimize" their representation for that particular task (i.e.: form a representation for which tractable solutions exist in that context, rather than using the same kind of representation across all contexts in which that problem occurs).

However, there are obvious interactions between the structure of our representations

and the cost of manipulating these representations via particular algorithms. Furthermore, by drawing on insights from parameterized complexity analysis, we can see that the cost of using two different algorithms might not "scale up" along the same dimensions. That is, if we have a set K of parameters that characterize a space of representations, and two different algorithms A_1 and A_2 for manipulating those representations, there may be cases (as we shall demonstrate in section 3.3) where parameter $k_1 \in K$ causes intractability in A_1, while a different parameter $k_2 \in K$ causes intractability in A_2. Thus, choosing an optimal algorithm depends on the structure of our representations, but choosing the optimal representation also depends on the set of algorithms we can use to manipulate it. Thus, we have a bi-directional interaction between the choice of how to represent uncertainty in a particular context, and the optimal choice of algorithm for manipulating that representation. For this reason, we advocate for a more pluralistic approach to boundedly rational cognitive modeling, in which we consider simultaneously optimizing over a joint space of representations and algorithms for manipulating those representations. In the next section, we will sketch out the general requirements of a framework for performing this analysis, point to some existing formal tools that are well-suited for such a framework, and demonstrate how, even in a simple case with a fairly restricted space of options, this optimization can be highly non-trivial.

3 Framework sketch

The analysis we outline in the previous section requires two core components. The first is a unified framework for formalizing both a space of possible representations of a task, and a space possible algorithms for manipulating these representations. As we show in the following section, Probabilistic Programming Languages (PPLs) are a particularly

well suited for this purpose (Goodman 2013), and are compatible with the assumptions underlying much of the current literature on bounded rationality. The second component is a methodology for computing a cost profile for each algorithm as a function of the representation to which it is applied. Drawing on parameterized complexity theory, the aim is to identify a set of relevant dimensions that characterize the different representations within this space, and compute the cost of each algorithm in terms of these dimensions. This will enable a joint optimization over representations *and* algorithms. As we shall argue in section 4, a full analysis would require dimensions of comparison not typically addressed in the current literature (e.g.: expectations over future data streams). However, we provide a case study in section 3.3 which demonstrates that, even when we restrict our analysis to a small set of relevant dimensions and a small space of possibilities, this optimality analysis can still be fairly non-trivial.

3.1 Probabilistic programming languages and generative models

A probabilistic programming language (PPL) extends a deterministic programming language with a set of stochastic primitive functions. For example, we can define a stochastic primitive $flip(w)$ that returns a 1 with probability w, or a 0 with probability $1 - w$, and a function $roll(n)$ which returns an integer between 1 and n uniformly at random. We can derive more complex functions from stochastic primitives via composition and recursion. For example, the program below simulates flipping a coin with bias w, then rolling a three-sided die if the flip comes up heads, or a six-sided die if the flip comes up tails:[3]

[3]For these examples, we use a condensed, intuitive psueodocode based on WebPPL, a probabilistic programming language for generative models (Goodman et al 2016)

```
flip_and_roll(w){
    f = flip(w)
    if (f == 1) {return roll(3)} else {return roll(6)}
}
```

Note that, as a probabilistic program, repeated calls to $flip\_and\_roll(w)$ with the same input value will result in a distribution of different output values. However, the PPL contains an operator that enables analytic computations of these probabilities as well. In particular, for a stochastic primitive function f and a value x in its range, the operator $Prob(f, x)$ returns the probability that f will output x. Thus, given a stochastic primitive function f, we can analytically compute the distribution it encodes by applying $Prob(f, x)$ to each x in its range, or we can approximate this distribution by running f repeatedly on the same input and tabulating the frequency of each output. These two basic operators enable a range of methods for computing or approximating more complex distributions. For a purely analytical computation, we can enumerate each possible execution history of the program and multiply the probabilities associated with each primitive decision (see Figure 2a for an example), while a purely stochastic approximation simply requires repeatedly running the program and tabulating its outputs. This also enables a range of intermediate algorithms, by applying the analytical operator $Prob(f, x)$ to certain primitive random decisions, while approximating other random decisions via sampling.

While this enables a range of algorithms for computing (or approximating) a distribution over *outputs*, most inference problems of interest involve further manipulation of this distribution. Suppose, for example, that a program involves some set X of random variables, including a subset $E \subset X$ that we get to observe, and a subset $Q \subset X$ that we don't get to observe. In order to reason about the value of the

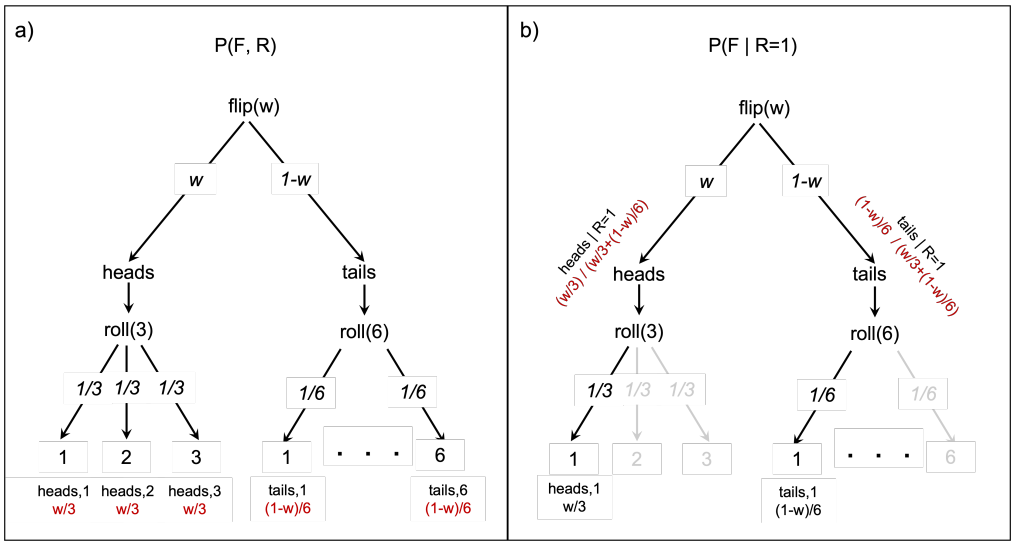

Figure 2: Diagram of procedure for computing the distributions implied by probabilistic programs. Panel a) depicts the procedure for computing the full joint distribution over all of model variables. Panel b) depicts procedure for computing a conditional distribution, given the observed value of one model variable. The probabilities corresponding to the target distribution are highlighted in red.

latent variables Q given the observations E, we need access to the posterior distribution $P(Q|E)$. With the *flip_and_roll* function, for example, suppose we observe that the die roll R resulted in a value of 1, and we must infer the result of the initial coin-flip F. At a computational level, this corresponds to computing the distribution $P(F|R = 1)$. At the algorithmic level, the basic operators of the PPL enable multiple ways to compute or approximate the target distribution.

To compute the distribution analytically, we can enumerate each possible execution history of the program, multiplying the weights of the primitive distribution at each random choice, and omitting any execution history which violates the observations (in this case, any history with a die roll not equal to 1). The conditional probability that the coin flip was heads, given that the die roll was 1, is equal to the total probability mass of all (non-excluded) executions where the coin flip was heads, divided by the total probability mass of all executions where the die roll was 1 (see Figure 2b). This

procedure allows the system to analytically compute any conditional distribution from any probabilistic program (with finite and discrete outputs).

At the other extreme, we can generate unbiased samples from the posterior distribution $P(Q|E)$ using a straightforward technique called rejection sampling: we simply run the program repeatedly until it outputs a value that matches the observation (in this case, until it results in a die roll of 1), then return the value of the query variable (in this case, the coin flip) associated with the final execution. Note that the analytic procedure uses only deterministic computation, but involves no random decisions, while rejection sampling may involve many random decisions, but involves no deterministic computation. We can therefore interpret these two algorithms as opposite endpoints of a spectrum, with fully deterministic solutions at one extreme and fully stochastic solutions on the the other. Between these two endpoints lie a range of intermediate methods that can be implemented using the same core machinery, including particle filters and various Markov Chain Monte Carlo (MCMC) methods. These methods use a mix of both random decisions and deterministic computation to generate (usually biased and autocorrelated) samples from the target distribution $P(Q|E)$. This trade-off between deterministic computation and random decisions is just one of several possible dimensions relevant to our analysis, but we shall focus on this dimension for our case study in section 3.3, to demonstrate that this trade-off alone entails a non-trivial optimization problem, even within a fairly restricted problem space.

3.2 Rational representations of uncertainty

The previous section demonstrates how PPLs can simultaneously encode a particular representation (i.e.: generative model) of a problem, and a set of algorithms for manipulating that representation. It is clear, however, that given a set of stochastic primitives and arbitrary recursion, we can define a rich space of possible representations

for the same problem. We therefore need some a systematic way of comparing these possible representations. One way of evaluating potential representations is to contrast "task-dependent" or "opportunistic" representations, which only represent uncertainty in the decision variable itself, with "constitutive" representations, which explicitly represent not just the decision variable, but a slew of latent variables thought to be involved in the causal process that generates the decision variable. Consider our earlier example, in which we observe an agent's initial movements $x_1, \ldots, x_{t-1}$ through some environment W, and must then predict the agent's next move x_t. A fully task-dependent representation for this task would represent a probability distribution over x_t as a direct function of the input variables $W, x_1, \ldots, x_{t-1}$ (Figure 3a). Intuitively, this representation encodes an assumption that the agent follows some fixed "script," such that the probability of the next action x_t is directly implied by the previous actions.

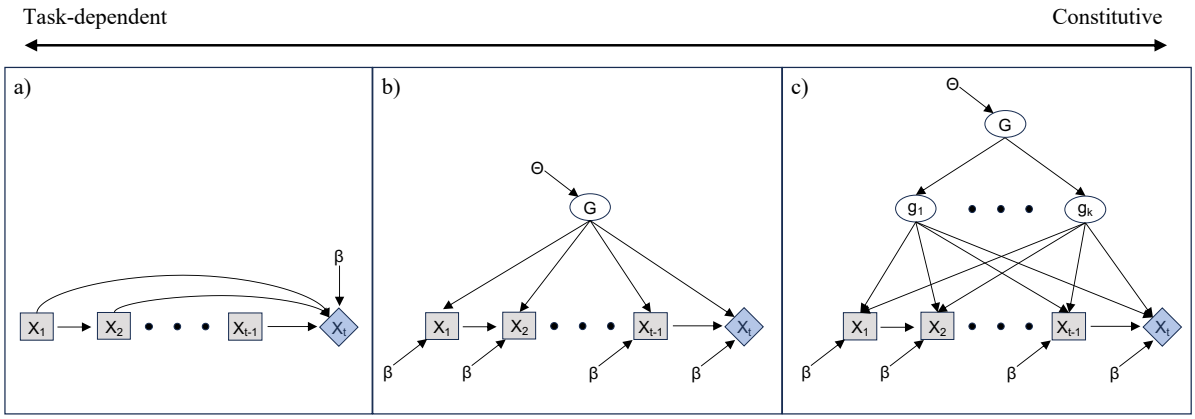

Figure 3: Examples of possible representations for an action-prediction task. Variables shaded in grey are observed (in this case, the agent's previous actions). Variable shaded in blue is the target of prediction (agent's next action). Variables in ovals are latent states (e.g.: goals). Variables without borders are parameters that encode the relevant probability distributions (e.g.: probability of taking an action given previous action and goal). Panel a) depicts a fully task-dependent model that only represents uncertainty in the target variable. Panel b) depicts a simple mentalistic model that posits a single latent goal state. Panel c) depicts a more complex mentalistic model that posits a hierarchical goal state with planning over sub-goals.

A constitutive representation, on the other hand, explicitly represents the latent mental states that are thought to cause the agent's behavior. For example, Figure 3c shows a complex goal model that posits a high-level goal G (e.g.: a certain recipe), which entails a set of sub-goals $g_1, \ldots, g_k$ that are required to fulfill the high-level goal G (e.g.: a list of ingredients), which in turn determine the agent's plan (e.g.: path through the grocery store that meets all the required ingredients). Of course, this distinction between task-dependent and constitutive representations is a graded, rather than binary notion, and we might consider a range of intermediate representations that include certain latent states but omit others. For example, Figure 3b depicts an intermediate representation

that posits a single static goal or preference state. Intuitively, we might interpret this as encoding the assumption that the agent is following one of several possible "scripts," and the single goal variable encodes which script the agent is executing (e.g.: Davis & Jara-Ettinger 2022).

Given these different representations of the same task, how do we evaluate and compare them? One obvious dimension is the cost of manipulation: in general, richer and more constitutive representations are costlier to manipulate and compute than simpler, more task-dependent representations, though the exact rate at which these costs scale depends on the nature of the algorithms being used (e.g.: the number of deterministic computations versus random decisions required). On the other hand, task-dependent representations tend to be highly inflexible, requiring a distinct representation for each possible task, even within a similar context, while constitutive representations enable much greater generalization and flexibility (Koblinger et al 2021). In our action prediction example, although we could predict an agent's behavior by memorizing a set of scripts that they tend to follow, it is likely that those scripts would vary widely across contexts (e.g.: the scripts one follows in the grocery store are unlikely to be the same scripts one follows at an airport). On the other hand, if we represent the mental states that cause the agent's actions, we can generalize those mental states across contexts: knowing that the agent likes hamburgers improves our ability to predict the agent's behavior in both a grocery store and an airport. Other potentially relevant dimensions for comparison include the memory cost of storing the representation (in particular, how many independent parameters must be stored), and the amount of data required to effectively learn the representation. The current literature on bounded rationality has paid considerably less attention to these last three factors- generalizability, memory requirements, and learnability- though some recent work has begun to address the first two in this list [cite]. We will return to this in section 4, when

we consider how to expand the current scope of boundedly rational cognitive modeling to enable a proper optimality analysis over representations and algorithms.

3.3 Case study: action prediction

We now provide a simple case study to motivate what this analysis might look like. As described in the previous sections, a fully formalized framework for this joint optimization is beyond the scope of this paper, in part because it will necessarily involve dimensions of comparison that have been under-explored in the current literature (see section 4). Thus, rather than a full demonstration, our aim here is to show that, even with a fairly constrained and simplified problem, and even with only a single dimension of comparison (deterministic computation versus random decision-making), this optimization can still be quite non-trivial. To this end, we return to our action prediction example, where we observe an agent's environment W and first few steps $\bar{x} = x_1, \ldots, x_{t-1}$, and wish to predict the agent's next step x_t. Of course, there is a wide range of generative models that one could use to represent this task, and a wide range of algorithms one could use to manipulate these representations. We will restrict our analysis to the candidate models shown in Figure 3, and the two algorithms described in section 3.1: enumeration, which only involves deterministic computation and no random decisions, and rejection sampling, which only involves random decisions and no deterministic computation.

We shall start with the fully task-dependent model in Figure 3a: under this model, the probability that the agent will take action x_t is a direct function of the agent's previous steps and parameter β: $P(x_t|\bar{x};\beta)$. Importantly, we are assuming that the observer has already learned the representation and relevant parameters, so the cost of acquiring the representation is not currently a factor (though we will return to this in section 4), and we only consider the cost of manipulating this representation to solve the

task. For the fully task-dependent representation, the cost of manipulation is quite low- in fact, there is barely any computation required. For an analytic solution, the posterior probability $P(x_t|\bar{x};\beta)$ for any x_t is already stored in the parameter vector β, so computing this probability exactly involves a single step (essentially, looking up the corresponding probability). Similarly, we can obtain an unbiased sample from $P(x_t|\bar{x};\beta)$ with a single random decision. Thus, the fully task-dependent representation enables an extremely efficient solution that only requires a single computation or random decision. Of course, this is not the full story: in order to enable such efficient computation, we must store a significant number of independent parameter values in β (essentially, one vector for each possible sequence of previous actions $x_1, \ldots, x_{t-1}$). Furthermore, this representation is specific to one particular environment, so our knowledge of these parameters is unlikely to be of any use in a slightly different context.

On the other end of the spectrum, we shall now consider the cost of manipulating the complex goal model (Figure 3c) for solving this task.[4] For an unbounded observer with this representation, predicting the agent's next action requires marginalizing out the latent goal variables, i.e.:

$$P(x_t|\bar{x}) = \sum_{g_1,\ldots,g_k} P(x_t|x_{t-1}, g_1, \ldots, g_k) P(g_1, \ldots, g_k|\bar{x})$$

Intuitively, this requires computing, for each combination of sub-goals $g_1, \ldots, g_k$, the probability that the agent would take action x_t, given those sub-goals, weighted by the posterior probability that the agent has those sub-goals, given their prior behavior. The term $P(x_t|x_{t-1}, g_1, \ldots, g_k)$ is already encoded into the model by the parameter β, so the operation requires computing the goal posterior $P(g_1, \ldots, g_k|\bar{x})$ for each possible

[4]We omit the analysis for the intermediate model in Figure 3b, as this is simply a special case of Figure 3c where the number of sub-goals k is fixed to 1

combination of sub-goals. Thus, analytically computing $P(x_t|\bar{x})$ for a single value of x_t involves M^k total computations, where M is the number of values that each sub-goal variable g_i can assume. Thus, the cost of the fully deterministic solution using this representation grows exponentially in the number of sub-goals k.

Now we contrast this against the cost of obtaining a single unbiased sample of $P(x_t|\bar{x})$ via rejection sampling, which does not involve any deterministic computation, and only involves random decisions. Recall that rejection sampling involves repeatedly running the generative model "forward" until we obtain a sample for which the observable variables $x_1, \ldots, x_{t-1}$ have the same values that we have observed in $\bar{x}$. A single forward run of this model involves first sampling the sub-goal vector $g_1, \ldots, g_k$ from the goal prior Θ (k random decisions), then simulating the agent's behavior for $t-1$ steps ($t-1$ random decisions). Of course, we cannot precisely compute the number of samples required before we obtain one that matches our evidence. However, it is straightforward to compute that, on average, we should expect to generate $1/P(\bar{x})$ samples before we obtain one that matches, where $P(\bar{x})$ is the overall probability of the evidence $\bar{x}$. Thus, obtaining a single unbiased sample of $P(x_t|\bar{x})$ via rejection sampling requires an average of $1/P(\bar{x})$ samples from the forward model, each of which involves $k+t-1$ random decisions. Of course, this is not the full story, as the analytic solution is always guaranteed to provide the correct answer, while rejection sampling only gives us a single unbiased sample from the predictive distribution $P(x_t|\bar{x})$. The full analysis would require some measure for the value of accuracy- that is, how much does it cost to get the wrong answer? Thus, we cannot properly establish which strategy is optimal without this extra piece of information.

However, the present analysis is still sufficient to draw some useful conclusions: first, while the cost of an analytically computing $P(x_t|\bar{x})$ grows exponentially in the number of sub-goal states k, the cost of generating an unbiased sample from $P(x_t|\bar{x})$ is only

polynomial in the number of sub-goals. On the other hand, the cost of an unbiased sample grows exponentially in the surprisal of the evidence (i.e.: $-log(P(\bar{x}))$, while the cost of an analytic solution is completely independent of this value. Thus, even within this fairly restricted example, we see an important trade-off emerge: as our representations become richer and involve a larger number of interconnected latent variables, the cost of analytic computations grows exponentially, while the cost of generating unbiased samples only grows in polynomial time. This suggests that deeper and richer representations may be most efficiently manipulated via algorithms that rely more heavily on random decisions, while flatter and simpler representations enable more efficient analytic solutions.

4 What's missing?

The previous section outlines the general requirements for the sort of analysis framework we advocate, and demonstrates that, even with a simplified toy problem, there are non-trivial interactions between how we represent the uncertainty in that problem (e.g.: the complexity of the generative model) and the optimal way to manipulate that representation (e.g.: via deterministic computation or unbiased sampling). However, it should be clear that the analysis in this simple demonstration is not the full story. After all, our analysis showed that the cost of computing the solution via the fully task-dependent representation is constant, so we might expect a resource-rational observer to always form fully task-dependent representations. However, this would fly in the face of both the vast theoretical literature on how humans represent complex causal systems [cite], as well as a growing empirical literature suggesting a broader flexibility in how people represent uncertainty [cite]. Thus, in this section, we consider what other factors are relevant for optimizing over representations.

4.1 Memory

While much of the existing literature on resource-rationality has focused on computation time, another important cognitive constraint is memory. In the context of our previous example, we can interpret this constraint in terms of the number of independent parameter values that must be maintained in order to store a model. In general, we can usually trade off computation for memory, by simply storing the output of a particular computation as a fixed parameter value. On an intuitive level, the fully task-dependent model is only able to achieve such computational efficiency by making an extreme trade-off: rather than relying on internal computations to predict the probability of an action given previous actions, this model simply stores all of these probabilities as fixed parameter values. Thus, in order to maintain this representation for a particular task, one must store a separate independent parameter vector for each possible state-sequence $x_1, \ldots, x_{t-1}$. Depending on the context, this might be completely feasible. Suppose, for example, that we are watching an agent navigate a very small grocery store with only three stands (say, a deli, a produce stand, and a dairy stand). In this case, given enough evidence of the agent's past behavior (see 4.3), it may be perfectly sufficient to simply memorize the order in which the agent typically visits these three stands.

Outside of a highly constrained environment, however, this strategy would likely impose a prohibitive memory requirement, making a fully task-dependent representation infeasible. This highlights the first major benefit of a more constitutive representation: by leveraging a richer representation of the latent variables and processes that generate observable behavior, we can drastically reduce the number of independent parameter values that must be stored. For example, the complex goal model in Figure 3c requires a single parameter vector for the agent's goals (e.g.: the agent's preferences over possible goal states), and a single parameter that captures the agent's degree of "noisiness" when executing a plan (e.g.: how deterministically they follow the most efficient path). Thus,

richer and more constitutive representations can drastically decrease the memory requirements for storing a representation at the expense of increased computation costs for manipulating the representation. This introduces another relevant trade-off for optimizing representations: do we dedicate more memory resources to store a representation that costs less to manipulate, or do we accept a higher cost of manipulation in order to economize our memory requirements?

4.2 Generalizability

A second drawback of task-dependent representations that the example from section 3.3 failed to highlight is generalizability. As the name suggests, a task-dependent representation is specific to a particular task or context: it encodes the uncertainty in a particular decision variable (e.g.: an agent's next action) as a direct function of the observable variables in that context (e.g.: the agent's previous actions). While this can enable efficient solutions for one particular context, a task-dependent representation is not well-suited for generalizing beyond that context. For example, if we simply memorize a distribution of scripts that a particular agent tends to follow in a particular environment, those scripts are unlikely to provide any predictive power in a very different environment. Thus, simply memorizing that a particular agent tends to visit the deli counter, the dairy stand, and the produce stand in that order at a particular grocery store does not help us predict how that agent will navigate, say, a mall food court- we would likely have to learn and memorize an entirely different set of scripts for that context. Of course, the feasibility of this strategy depends on the distribution of tasks that we expect to face: if we only need to predict the actions of a small set of agents across a small set of similar environments, it may be more efficient to simply memorize a script for each agent in each context.

On the other hand, a constitutive representation, which explicitly represents the

latent causal processes that generate observable behavior, is much better suited for generalizing beyond a particular context. Rather than simply representing, say, the most likely trajectories of an agent through a particular environment, if we represent an agent's persistent mental states (e.g.: preferences), as well as the causal process that links those mental states to observable behavior, we can leverage the same representation to make predictions across multiple tasks. Thus, while explicitly representing how an agent traverses a grocery store doesn't help us predict their movement through a mall food court, representing an agent's preference for ham and cheese sandwiches, and how those preferences influence the agent's behavior, allows us to make useful predictions in both the grocery store and the food court. This introduces another trade-off to the equation: do we maintain a larger number of more task-specific representations, or a smaller number of richer and more flexible representations?

4.3 Learnability

It is important to note that, in the example from section 3.3, we assumed that the observer already had access to fully parameterized versions of each representation. Thus, this analysis glossed over the cost of *learning* these representations in the first place (i.e.: inferring the values of relevant parameters). In reality, however, the time it takes to learn a representation may be a significant factor in deciding how to represent uncertainty in a task. In general, richer and more constitutive representations can be learned more quickly, and often from less data, than fully-task dependent representations, a phenomenon sometimes referred to as the "blessing of abstraction" [cite]. There are two high-level reasons for this difference, the first of which is closely related to generalizability: in a fully task-dependent representation, the only data relevant for learning the representation are data observed in same context being represented. For example, if we represent an agent's trajectories through an environment

as an explicit distribution over trajectories, the only data useful for inferring that distribution are observations of the trajectories themselves. Thus, we can only learn that an agent generally visits the deli, produce stand, and dairy counter in that order by repeatedly observing that agent traverse that particular grocery store. On the other hand, if we represent uncertainty in terms of the agent's mental states, we can integrate information from across multiple contexts to learn the relevant parameter values. For example, if we explicitly represent an agent's preferences (rather than a direct distribution over trajectories), we can leverage data from multiple contexts (e.g.: any context in which the agent makes a choice of what to eat) to infer the parameter values that capture the agent's preferences.

A second difference in learning these representations is that learning a task-dependent representation requires labelled data (e.g.: observation of the agent's full trajectory), while richer and more constitutive representations can leverage unlabelled data as well (Koblinger et al 2021). This is due to the fact that a constitutive model can be used to estimate the missing labels. For example, suppose we observe the agent go to the dairy counter, then take two more steps in another direction, but we don't get to observe the rest of the trip. Using a mentalistic goal model, an observer could infer a posterior distribution over the agent's possible goals, based on the partial trajectory, then use that inferred goal distribution to predict the probability of each possible next step. The observer could then update the parameters in the model by averaging over all possible completions of the trajectory, weighted by the posterior probability of that trajectory. Intuitively, this means that the observer can use the latent causal processes encoded in the representation to simulate the missing portion of the data, and use that simulated data to perform additional learning. Thus, even though task-dependent representations are more efficient to manipulate, they generally require more data, more specific data, and more labelled data in order to learn, compared to constitutive

representations. Thus, choosing the optimal representation may depend in part on our expectations about the availability and cost of future data.

4.4 Putting it all together: the value of representation

While a full specification of a framework that unifies all of these trade-offs is beyond the scope of this paper, we can motivate how this might be done by looking toward some recent work. In a standard approach to resource rational analysis, we consider some task t, and a set $\mathcal{A}$ of possible algorithms for solving the task. These algorithms may differ in both the reward they yield if applied to this task (e.g.: how accurately or consistently they produce the right answer), which we can denote by $R(a,t)$ for $a \in \mathcal{A}$, as well as the cost of implementing the algorithm in the task, which we can denote by $C(a,t)$. The overall utility derived from applying algorithm a to task t is thus $U(a,t) = R(a,t) - C(a,t)$, and the resource-optimal algorithm for solving the task is defined as $a^* = argmax_a U(a,t)$ [cite]. In many cases, however, we might have some uncertainty about which exact task we will face (e.g.: which shop an agent is navigating, or which agent is navigating the shop). If we have some belief about the set $\mathcal{T}$ of possible tasks, and the probability $P(t)$ that we will face a certain task $t \in T$, we can compute the *expected* utility of applying algorithm a as

$$EU_P(a) = \sum_t U(a,t)P(t) \tag{4.1}$$

and define the optimal algorithm as $a^* = argmax_a EU_P(a)$.

In this context, our proposal essentially amounts to generalizing this optimization to a space $\mathcal{R}$ of possible representations tasks in $\mathcal{T}$, *and* a set $\mathcal{A}$ of algorithms for manipulating those representations. For a fixed representation $r \in \mathcal{R}$, we can extend the above definitions to $R(a,r,t)$, $C(a,r,t)$, and $U(a,r,t)$, to respectively define the reward,

cost, and overall utility of applying algorithm a to representation r for task t. We can then define the overall utility of representation r for solving task t as $U(r,t) = max_a U(a,r,t)$ (i.e.: the utility obtained by applying the best algorithm for that representation). Similarly, for a given distribution over tasks $P(t)$, we can define the expected utility of a representation for solving those tasks as

$$EU_P(r) = \sum_t U(r,t)P(t) \tag{4.2}$$

Note that the generalizability of a representation only becomes relevant when we consider the full distribution of tasks we might encounter, as in equation 4.2. If $P(t)$ is highly concentrated on a small set of similar tasks, it may be more efficient to use a cheaper, task-dependent representation. On the other hand, if $P(t)$ has non-trivial support over a large set of different tasks, we may require a richer but more generalizable representation to adequately solve those tasks.

Finally, note that in the standard usage, the cost function $C(a,r,t)$ only captures the cost of a implementation- that is, the cost of applying a to r to solve a single instance of t. However, two of the "costs" described in the previous section apply outside the scope of a particular implementation: the cost of maintaining the representation r in memory (i.e.: the number of independent parameter values required), and the cost of learning the representation (i.e.: inferring the values of those parameters). These costs are specific to the representation itself, and are independent of the algorithm used to manipulate that representation, or the task(s) for which the representation applies. Thus, when computing the expected utility of the representation, these costs would appear outside the scope of the expectation operator, i.e.:

$$EU_P(r) = \left(\sum_t U(r,t)P(t) \right) - C_{mem}(r) - C_{learn}(r) \tag{4.3}$$

A remaining challenge is determining how to formalize these two additional costs in a way that can be directly compared to (and subtracted from) the overall utility of the representation. Intuitively, computational costs are typically formulated as a measure of time (i.e.: how long it takes to solve a problem), whereas memory constraints are spatial. Furthermore, while the cost of learning the representation can also be formulated in terms of time, the actual cost of learning a representation will depend on our expectations about the availability and cost of future data. For example, while a fully task-dependent representation for action prediction may require a large quantity of data to learn, if we know that we will have access to a large quantity of relevant data (e.g.: CCTV footage of the agent's path through the grocery store over many days), then it may be feasible to efficiently learn that representation. On the other hand, if we expect to only have access to a handful of observations of the agent's choices, perhaps from several different environments, it may be prohibitively expensive to learn the task-dependent representation, but perfectly feasible to learn the relevant preference parameters for a constitutive representation. Thus, while a fully formalized framework will require further investigation, equation 4.3 sketches out how we might usefully define a joint utility function over representations and algorithms that a resource-rational agent could optimize.

5 Discussion and future work

Bounded rationality is a promising research program that resolves a longstanding tension in cognitive science and psychology. On the one hand, there are many contexts in which human judgments appear to reflect (approximately) rational statistical inference. Indeed, over the past two decades, rationalist cognitive models have been used to provide computational-level accounts for nearly every aspect of human cognition (Griffiths et al

2008, Oaksford & Chater 2007). On the other hand, a computation-level rational analysis does not explain how people are able to perform these seemingly intractable computations, nor does it explain the seemingly sub-rational biases and errors we systematically display across a wide range of inference tasks (Tversky & Kahneman 1974, Epley & Gilovich 2006, Lichenstein et al 1978, Mozer et al 2008). Boundedly rational analysis seeks to resolve this tension by considering the limited cognitive resources with which real-world human minds operate, and demonstrating that our apparently sub-rational biases and heuristics actually seem to reflect the rational allocation of limited resources.

That said, it is difficult to determine the appropriate scope of focus for boundedly rational cognitive models. The most common approach is to characterize an inference problem and its optimal solution at the computational-level, then consider algorithms for approximating that that solution. As we and others have argued, however, this approach is neither immediately demanded nor immediately justified by the assumptions of bounded rationality. First, approximation does not, in general, make intractable problems tractable: in many cases, approximate solutions can be just as prohibitively expensive as exact solutions (Kwisthout et al 2011). Even in cases where approximation *is* tractable, there is no general guarantee that approximating the optimal solution is more rational or efficient than some other context-specific heuristic (Icard 2018). Furthermore, there are many different ways that an agent could represent uncertainty in a given task, and many different algorithms for manipulating those representations, all of which could be implemented using the same cognitive machinery typically assumed by these models. Thus, focusing on the optimal use of one particular type of algorithm for approximating the ideal solution may unnecessarily limit our search for plausible cognitive models, and often lacks the rationalist justification that motivated the search in the first place.

For this reason, we advocate for a pluralistic approach to boundedly rational cognitive modeling, where we consider the representational and computational primitives to which an agent has access, and optimizes over the full space of representations and manipulations that could be implemented with those primitives. We demonstrated how, even in a simple case study, there are non-trivial interactions between the way we represent uncertainty (e.g.: the richness of the latent structure encoded in our representations) and cost of manipulating those representations via different algorithms (e.g.: via exact enumeration or unbiased sampling). We further argued that, by optimizing over representations *and* algorithms, we introduce additional dimensions of trade-offs such as the static memory requirements to maintain a representation, the degree to which a representation generalizes across tasks, and our expectations about the availability of data relevant for learning the representation. While some very recent work has begun to consider the joint optimization problem over representations of a task *and* algorithms for manipulating the representations (e.g.: [cite]), these additional dimensions remain under-explored in the current literature. We further highlighted some existing formal tools that are well-suited for this purpose, and outlined how a unified framework for this optimization might be constructed. That said, this work points to some important future research, both theoretical and empirical.

On the theoretical side, our notion of "value of representation" will require some additional work to fully formalize and implement. In particular, integrating memory constraints into a unified notion of cost may be challenging, in part because memory can impose different kinds of constraints depending on the type of representations being used. When maximizing a posterior distribution analytically, for example, even though the agent must compute the posterior probability for each possible answer, they need only *remember* one possible answer at a time- if a new answer is determined to have a higher posterior probability than the previously remembered answer, the agent is free to

"forget" the previous answer and only retain the new one. On the other hand, if the agent is, say, approximating the distribution with a set of samples, then the agent's memory limitation will directly constrain the number of samples the agent can retain, and thus the accuracy of the approximation. In future work, it will be important to consider how memory limitations fit into this broader analysis framework, and some recent work has already begun to investigate this issue [cite]. Additionally, the cost of *learning* a representation may be highly dynamic and depend on our knowledge or expectations about the distribution of environments we will face, and the availability (and cost) of relevant data in those environments.

On the empirical side, a more pluralistic approach to bounded rationality suggests several new directions for behavioral studies. First, although there is some behavioral and neural evidence that people can represent uncertainty in multiple ways (e.g.: [cite]), there is little to no empirical work (as far as we are aware) that explores how flexibly people can *adjust* their representations in response to specific task demands or environments. Based on the arguments presented here, we might expect, for example, that people will form more task-dependent representations when they expect to only face a small set of similar environments, or when they expect to have access to large quantities of relevant behavioral data, but will leverage more constitutive representations when they expect a wider range of dissimilar environments, or a paucity of relevant behavioral data. Koblinger et al (2021) outline a general approach to behavioral studies into the task-specifity of people's cognitive representations, and a similar approach could be leveraged to investigate the flexibility of those representations across different task environments.

Furthermore, these insights may lead to novel predictions about when we expect people to rely on sampling-based approximations versus exact computation. The case study we presented in section 3.3 suggests such a study: if, as current work suggests, the

variability in human responses reflects an underlying sampling process, then an agent who solves a problem exactly should display significantly less variance in their responses than an agent who approximates a solution via sampling. We can therefore leverage this principle to derive testable hypotheses about how people manipulate different representations of uncertainty. Some work has already demonstrated that people can be motivated to make more accurate inferences with less variability by increasing the potential payout of a correct answer (Vul et al 2014) or increasing the noisiness of a stimulus (Hamrick et al 2015). Given the principle that richer representations are more costly to manipulate analytically, it should also be the case that increasing certain parameters of an inference problem (e.g.: the number of possible hidden states or the likelihood of an observation) beyond a certain threshold should induce a switch from approximation to exact computation, or vice versa. Such a switch would be characterized by a sharp increase or decrease in response variability as a problem moves above or below one of these thresholds. Thus, theoretical development of this framework will both necessitate and generate a plethora of novel behavioral studies.

References

[1] Anderson, J. R. (1990). The Adaptive Character of Thought. *Psychology Press*.

[2] Andrieu, C., De Freitas, N., Doucet, A., & Jordan, M. I. (2003). An introduction to MCMC for machine learning. *Machine learning*, 50(1), 5-43.

[3] Baker, C., Saxe, R., & Tenenbaum, J. (2011). Bayesian theory of mind: Modeling joint belief-desire attribution. In *Proceedings of the annual meeting of the cognitive science society* (Vol. 33, No. 33).

[4] Baker, C. L., Saxe, R., & Tenenbaum, J. B. (2009). Action understanding as inverse planning. *Cognition*, 113(3), 329-349.

[5] Bonawitz, E., Denison, S., Gopnik, A., & Griffiths, T. L. (2014). *Win-Stay, Lose-Sample: A simple sequential algorithm for approximating Bayesian inference.* Cognitive psychology, 74, 35-65.

[6] Buesing, L., Bill, J., Nessler, B., & Maass, W. (2011). Neural dynamics as sampling: a model for stochastic computation in recurrent networks of spiking neurons. *PLoS computational biology*, 7(11), e1002211.

[7] Dasgupta, I., Schulz, E., & Gershman, S. J. (2017). Where do hypotheses come from?. *Cognitive psychology*, 96, 1-25.

[8] Davis, I., & Jara-Ettinger, J. (2022). Hierarchical task knowledge constrains and simplifies action understanding. *Proceedings of the Annual Meeting of the Cognitive Science Society*, 44 (44)

[9] De Finetti, B. (1937). Le prevision: ses lois logiques, ses sources subjectives. In *Annales de l'institut Henri Poincare* (Vol. 7, No. 1, pp. 1-68).

[10] Denison, S., Bonawitz, E., Gopnik, A., & Griffiths, T. L. (2013). Rational variability in children's causal inferences: The sampling hypothesis. *Cognition*, 126(2), 285-300.

[11] Downey, R. G., & Fellows, M. R. (2012). Parameterized complexity. Springer Science & Business Media.

[12] Eberhardt, F., & Danks, D. (2011). Confirmation in the cognitive sciences: The problematic case of Bayesian models. *Minds and Machines*, 21(3), 389-410.

[13] Epley, N., & Gilovich, T. (2006). The anchoring-and-adjustment heuristic: Why the adjustments are insufficient. *Psychological science*, 17(4), 311-318.

[14] Gergely, G., Nadasdy, Z., Csibra, G., & Biro, S. (1995). Taking the intentional stance at 12 months of age. Cognition, 56(2), 165-193.

[15] Gerstenberg, T., Goodman, N. D., Lagnado, D. A., & Tenenbaum, J. B. (2021). A counterfactual simulation model of causal judgments for physical events. *Psychological Review*.

[16] Goodman, N. D. (2013). The principles and practice of probabilistic programming. *ACM SIGPLAN Notices, 48(1)*, 399-402.

[17] Goodman, N. D., & Frank, M. C. (2016). Pragmatic language interpretation as probabilistic inference. *Trends in cognitive sciences*, 20(11), 818-8

[18] Goodman, N.D., Tenenbaum, J.B., & The ProbMods Contributors (2016). Probabilistic Models of Cognition (2nd ed.) https://probmods.org/

[19] Gopnik, A., & Wellman, H. M. (2012). Reconstructing constructivism: Causal models, Bayesian learning mechanisms, and the theory theory. *Psychological bulletin*, 138(6), 1085.

[20] Griffiths, T. L., & Ghahramani, Z. (2005). Infinite latent feature models and the Indian buffet process. In NIPS (Vol. 18, pp. 475-482).

[21] Griffiths, T. L., Kemp, C., & Tenenbaum, J.B. (2008). Bayesian models of cognition. In *The Cambridge handbook of computational psychology* (R. Sun, ed.), ch. 3, 59-100, Cambridge University Press.

[22] Hamrick, J. B., Smith, K. A., Griffiths, T. L., & Vul, E. (2015). Think again? The amount of mental simulation tracks uncertainty in the outcome. In *Proceedings of the 37th annual conference of the Cognitive Science society*.

[23] Huttegger, S. M. (2013). In defense of reflection. *Philosophy of Science*, 80(3), 413-433.

[24] Icard, T. (2016). *Subjective probability as sampling propensity.* Review of Philosophy and Psychology, 7(4), 863-903.

[25] Icard, T. (2018). Bayes, bounds, and rational analysis. *Philosophy of Science*, 85(1), 79-101.

[26] Icard, T. (2021). Why be random?. *Mind*, 130(517), 111-139.

[27] Jara-Ettinger, J., Gweon, H., Schulz, L. E., & Tenenbaum, J. B. (2016). The naïve utility calculus: Computational principles underlying commonsense psychology. *Trends in cognitive sciences*, 20(8), 589-604.

[28] Jones, M., & Love, B. C. (2011). Bayesian fundamentalism or enlightenment? On the explanatory status and theoretical contributions of Bayesian models of cognition. Behavioral and brain sciences, 34(4), 169.

[29] Kersten, D., Mamassian, P., & Yuille, A. (2004). Object perception as Bayesian inference. *Annu. Rev. Psychol.*, 55, 271-304.

[30] Knill, D. C., & Pouget, A. (2004). The Bayesian brain: the role of uncertainty in neural coding and computation. *TRENDS in Neurosciences*, 27(12), 712-719.

[31] Koblinger, Á., Fiser, J., & Lengyel, M. (2021). Representations of uncertainty: where art thou?. *Current Opinion in Behavioral Sciences*, 38, 150-162.

[32] Kwisthout, J., Wareham, T., & van Rooij, I. (2011). Bayesian intractability is not an ailment that approximation can cure. *Cogn. Sci.*, 35(5), 779-784.

[33] Lichtenstein, S., Slovic, P., Fischhoff, B., Layman, M., & Combs, B. (1978). Judged frequency of lethal events. *Journal of experimental psychology: Human learning and memory*, 4(6), 551.

[34] Lieder, F., & Griffiths, T. L. (2020). Resource-rational analysis: Understanding human cognition as the optimal use of limited computational resources. *Behavioral and Brain Sciences*, 43.

[35] Lieder, F., Griffiths, T. L., & Hsu, M. (2018). Overrepresentation of extreme events in decision making reflects rational use of cognitive resources. *Psychological review*, 125(1), 1.

[36] Lieder, F., Griffiths, T., & Goodman, N. (2012). *Burn-in, bias, and the rationality of anchoring*. Advances in neural information processing systems, 25, 2690-2798.

[37] Ma, W. J., Beck, J. M., Latham, P. E., & Pouget, A. (2006). Bayesian inference with probabilistic population codes. *Nature neuroscience*, 9(11), 1432-1438.

[38] Marr, D. (1982). Vision: A computational investigation into the human representation and processing of visual information. San Francisco: W. H. Freeman.

[39] Milli, S., Lieder, F., & Griffiths, T. L. (2021). A rational reinterpretation of dual-process theories. *Cognition*, 217, 104881.

[40] Moreno-Bote, R., Knill, D. C., & Pouget, A. (2011). Bayesian sampling in visual perception. *Proceedings of the National Academy of Sciences*, 108(30), 12491-12496.

[41] Mozer, M. C., Pashler, H., & Homaei, H. (2008). Optimal predictions in everyday cognition: The wisdom of individuals or crowds? Cognitive science, 32(7), 1133-1147.

[42] Oaksford, M., & Chater, N. (2007). Bayesian rationality: The probabilistic approach to human reasoning. *Oxford University Press*.

[43] Pylyshyn, Z. W. (1980). Computation and cognition: Issues in the foundations of cognitive science. *Behavioral and Brain Sciences*, 3(1), 111-132.

[44] Robert, C. P., & Casella, G. (1999). The Metropolis-Hastings Algorithm. In Monte Carlo Statistical Methods (pp. 231-283). Springer, New York, NY.

[45] Salakhutdinov, R., Tenenbaum, J., & Torralba, A. (2012, June). One-shot learning with a hierarchical nonparametric bayesian model. In *Proceedings of ICML Workshop on Unsupervised and Transfer Learning* (pp. 195-206). JMLR Workshop and Conference Proceedings.

[46] Simon, H. A. (1980). Bounded rationality. In *Utility and probability* (pp. 15-18). Palgrave Macmillan, London.

[47] Simon, H. (1955). A behavioral model of bounded rationality. Quarterly Journal of Economics, 69(1), 99-118.

[48] Smith, K. A., & Vul, E. (2013). Sources of uncertainty in intuitive physics. *Topics in cognitive science*, 5(1), 185-199.

[49] Tversky, A., & Kahneman, D. (1974). Judgment under uncertainty: heuristics and biases. *Science*, 185(4157), 1124-1131.

[50] Vul, E., Goodman, N., Griffiths, T. L., & Tenenbaum, J. B. (2014). One and done? Optimal decisions from very few samples. *Cognitive science*, 38(4), 599-637.

[51] Woodward, A. L. (1998). Infants selectively encode the goal object of an actor's reach. *Cognition*, 69(1), 1-34.