ESTIMATING UTILITY FUNCTIONS FOR PREDICTABLY IRRATIONAL AGENTS WITH PREFERENCE CHANGE

PATRICK PAN

Abstract. This paper introduces an adaptive estimator for utility functions that accounts for agents exhibiting predictably irrational behavior through a Bayesian framework. Building upon and challenging the traditional von Neumann-Morgenstern utility theorem, the model I propose accommodates both errors in decision-making and dynamic changes in preferences, reflecting more realistic human behavior. Specifially, I utilize a sigmoid policy function following Hadfield-Mennell (2016)[1] to model the likelihood of an error being committed over a choice between two lotteries, and I model changes in an agent's utility using a Lévy Flight, a type of random walk also used to model financial instruments. In addition, I argue that both the vNM theorem and Ng and Russell's (2000) approach to modeling utility functions using Inverse Reinforcement Learning are significantly limited by their assumptions of time-invariant preferences and infallibility. Comparative simulations demonstrate that the adaptive estimator consistently outperforms an estimator that assumes ideal rationality as far as possible in accurately modeling utility functions.

[1] Hadfield-Menell et al., "The Off-Switch Game".

§1 Introduction: The vNM Utility Theorem and its Assumptions

AI value alignment, as defined by Stuart Russell, is the task of providing artificially intelligent systems with objectives that align with our own.[2] The current paradigm of AI alignment, based on reinforcement learning, uses the method of maximizing the expected value of a utility function which assigns higher values to good outcomes than bad ones, where 'good' and 'bad' are defined in a broadly moral sense. On many plausible theories of good outcomes, the satisfaction of at least some human desires or preferences contributes to the goodness of an outcome. Consequently, a significant portion of the task of AI alignment will consist in determining how to best satisfy human individuals' preferences, including under various kinds of uncertainty: both the uncertainty of what results an AI's action will produce, and the uncertainty of what human individuals' preferences are at any given time.

According to the von Neumann-Morgenstern (vNM) utility theorem, ideally rational agents whose preferences over lotteries satisfy the axioms of *completeness*, *transitivity*, *independence*, and *continuity* can be presented as maximizing a utility function that functionally represents those preferences.[3] A purely epistemic interpretation of the theorem does not posit that these agents actually represent or compute this utility function (either consciously or unconsciously), or that they intend to maximize anything at all. Rather, the theorem only shows that there exists some function U which assigns numerical utilities to outcomes such that the agent chooses outcome A over outcome B iff $U_A > U_B$—i.e., the utility of outcome A is strictly greater than the utility of outcome B.

If humans were indeed ideally rational agents, a certain harmony and simplicity would emerge in the task of AI alignment. An AI tasked with satisfying a certain human individual's preferences, among other things, might maximize a utility function that sums

[2] Russell, *Human Compatible: Artificial Intelligence and the Problem of Control*, ch. 5.
[3] Gustafsson, *Money-Pump Arguments*, ch. 1. .

several terms: one for the human's posited utility function, plus some other terms as well. The task of maximizing the overall utility function might proceed by first identifying the individual's utility function and then maximizing the resulting sum of that function and the other terms. (I will focus on a single-agent case at this point to avoid cases of optimally satisfying social preferences, which faces the problem of incomparability of utilities between agents.)

But while artificial agents powered by machine learning algorithms like reinforcement learning can be statistically shown to maximize their own utility functions under a specified range of circumstances, this assumption cannot be made about human agents in practice. The vNM theorem describes ideally rational agents, whose preferences satisfy the relevant axioms. Actual human agents diverge from this model of ideal rationality in at least three ways:

1. First, humans can violate the vNM axioms synchronically: for instance, by simultaneously affirming that apples are strictly preferred to bananas, which are strictly preferred to cherries, which are strictly preferred to apples.
2. Second, humans' preferences may change over time: I may strictly prefer apples to bananas today but not tomorrow, simply due to a change in my tastes.
3. Finally, humans also make mistakes when exhibiting preferences in behavior. Though I may strictly prefer apples to bananas, I might simply reach for the wrong fruit or utter the wrong word when offered a choice between the two.

It might be concluded from these three kinds of deviations from ideal rationality that the vNM theorem has no application in practice, and that we have no good reason to believe that humans are maximizers of expected utility. Most radically, Kripkensteinian skepticism about rule-following might suggest that no evidence could conclusively distinguish between a human having made a mistake on one hand and having adopted a different rule, i.e., maximizing a different utility function, on the other.[4] This would lead to a total indeterminism about what utility function one is actually maximizing at any given time. Thus, giving up all of the assumptions of ideal rationality without any replacement assumptions would mean not just that the vNM theorem's result is an 'approximation' in the way that, for instance, the ideal gas law approximates gases' behavior. Rather, the vNM would have no practical application whatsoever, providing no information at all about what utility function a human would be maximizing.

The goal of this paper is to develop an alternative method for modeling humans' utility functions that does not rely on the vNM theorem and its unrealistic assumptions. To do so, it will be necessary to make some alternate assumptions about humans' behavior and utility functions. This will allow constructing a Bayesian model for humans' utility functions that can be used to reconstruct utility functions from demonstrated preferences.

[4] Kripke, *Wittgenstein on Rules and Private Language.*

§2 Qualifying the Assumptions of the vNM Theorem

To counter this pessimistic conclusion that results from simply rejecting the vNM theorem's assumption of ideal rationality, I propose to consider a middle ground between the overidealized assumption that humans adhere completely and flawlessly to the vNM axioms and never change their utility functions on one hand, and the overly pessimistic assumption that humans violate the vNM axioms and commit errors in a totally unpredictable way and that their utility functions change unpredictably. We can independently accept or deny each of the following assumptions, and for assumptions which we deny, we can provide statistical models for the likelihood and nature of violations. I proceed to consider each assumption below.

§2.1 Humans' synchronic preferences obey the vNM axioms

Imperfect humans may obey some vNM axioms while not obeying others. Furthermore, we might model the degree to which any axiom is violated, both considering the number of violations relative to the number of exhibited preferences as well as the strength of the violation (e.g., even if $A \succ B \succ C \succ A$, it may matter whether U_A, U_B, and U_C are close to one another or not).

In this paper, I will assume that humans' 'true' preferences obey the vNM axioms synchronically, and that any apparent synchronic violations in choice behavior entails that the human has made an *error*, which I discuss below.

§2.2 Humans commit errors in predictable ways

I define an *error* as an instance in which we posit that some agent's utility function is such that $U_A > U_B$, but they choose *B* over *A*—that is, $B \succ A$. We could model the likelihood of an error being committed over a choice between *A* and *B* in terms of U_A and U_B; that is,

$$p(B \succ A \mid U_A > U_B) \underset{\text{df}}{=} \varepsilon(A, B) = \text{fn}(U_A, U_B)$$

Psychological evidence might provide intuitive hypotheses about f. On one hand, we might suppose that humans are more likely to commit an error the closer U_A and U_B are, or the closer to zero each are; mathematically, f correlates negatively with $|U_A|$, $|U_B|$, and $|U_A - U_B|$. This might be because, since the difference is less perceptible or significant, humans will not feel the need to spend much time deciding carefully. Though the rest of my behavior may indicate that my 'true' utility function ranks chocolate over vanilla ice cream, I might commit the error of choosing vanilla over chocolate because this decision is inconsequential. By contrast, if I am choosing between chocolate ice cream and the known sure loss of a limb, it would be exceedingly unlikely for me to make the mistake of choosing the latter.

On the other hand, we might believe that high stakes lead to stress and pressure, thus increasing the rate of error. Suppose the President of the United States is deciding whether to respond to reports of an enemy ICBM strike with a strike of the US's own on location A, location B, or no strike at all. Even if the president is appraised of all the relevant information by infallible and trusted advisors who show that the expected utility of one option (say, striking location B) is much higher than the alternative(s), we can imagine that the incredibly high stakes of the situation might impede the president's decision-making in a way that an ideally rational agent or automated decision-maker would not experience, given the same information. If this mechanism is the sole governor of error, then we might expect $\varepsilon(A, B)$ to correlate positively with $|U_A|$, $|U_B|$, and $|U_A - U_B|$.

Given this non-exhaustive list of psychological mechanisms that relate the likelihood of an error given the true utilities, we can construct error functions that depend in various ways on U_A and U_B as well as possibly other variables. Introducing any such models of errors achieves a middle ground between the unrealistic assumption that humans never make mistakes made by the original vNM theorem and the pessimistic assumption that humans make errors in ways that are not even describable in principle.

However, when preferences are sequentially exhibited, inconsistent preferences at different times are not necesssarily caused by errors: they may also evince *preference change.*

§2.3 Humans never change their utility functions.

Any sequence of preferences, even inconsistent ones ($A \succ B$ and $B \succ A$), can be explained by changes in utility functions. Thus, removing the constraint on constant utility functions without replacing this assumption leads to a total inability to describe what the true utility function is at any time in the past or predict what the utility function might be in the future.

However, our intuitive experience—and the clear ability for people to predict reasonably well what others will want in the future—suggests that, although preference change or transformative experiences do result in changes in one's posited utility function, such changes are either gradual and predictable, or large but infrequent. For instance, how much I like chocolate ice cream may slightly fluctuate day-to-day depending on my mood, but may change radically on rare occasions, such as after beginning a relationship with someone who loves chocolate or reading about the exploitative cacao industry.

One mathematical model that may be especially suitable for modeling time-structured preference change is the **Lévy Flight**. Lévy Flights are discrete-time random walks where the step sizes are independently sampled from a stable and (typically) heavy-tailed distribution. In other words, each step size is independent of the previous step size, and the step size is very likely to be small, but large steps are possible albeit increasingly (though less than exponentially) less probable as the step size increases. Lévy flights are often used to model financial instruments, where the price of the instrument on one day is likely to have changed little from the previous day, but large changes happen infrequently. By the same token, Lévy flights might be used to model human preference change.

To anticipate, the mathematical representation of preference change can thus be modeled as follows:

$$U_X(t) = U_X(t-1) + \Delta_X(t)$$

where $U_{X(t)}$ is the utility of lottery X at time t and $\Delta_X(t)$ is a random variable drawn from a stable and heavy-tailed distribution. If we remove the stability and heavy-tailed constraints on Δ_X, then this model of preference change would simply be a general random walk.

§2.4 Constructing Utility Functions for Predictably Irrational Agents

As seen in the preceding section, we can construct statistical models for the likelihood of committing errors and the change of utility functions over time. This allows us to produce a more sophisticated model for relating demonstrated preferences to utility functions than the proof of the vNM theorem utilizes. In Section 4, I will provide mathematical representations of the relationship between preferences and utility functions in four cases:

- original vNM, with ideally rational, infallible, constant-utility agents (Section 4.1)
- agents which violate assumption 2 above only (Section 4.2)
- agents which violate assumption 3 above only (Section 4.3)
- agents which violate both assumptions 2 and 3 (Section 4.4)

These representations will have a general form, where the error rate and/or changes in utility function is not concretely specified. In Section 4, I will specify error and utility-change functions to provide a more specific function, and in Section 5 I will explore some computational results from attempting to derive utility functions in these cases. However, I will briefly show in Section 3 how the existing model of *inverse reinforcement learning*, introduced by Ng and Russell,[5] does not explicitly model error and preference change.

[5] Ng and Russell, "Algorithms for Inverse Reinforcement Learning".

§3 The Limitations of Inverse Reinforcement Learning

Ng and Russell attempt to solve a problem similar to the one that I raise in this paper in their influential 2000 paper, "Algorithms for Inverse Reinforcement Learning".[6] There, they characterize the problem of *Inverse Reinforcement Learning* (or IRL) as follows:

Inverse Reinforcement Learning

Given 1) measurements of an agent's behavior over time, in a variety of circumstances, 2) if needed, measurements of the sensory inputs to that agent; 3) if available, a model of the environment.
Determine the reward function being optimized.

Ng and Russell remark, after introducing the problem, that IRL can be used as "computational models for animal and human learning", in which it is assumed that "in examining animal and human behavior we must consider the reward function as an unknown to be ascertained through empirical investigation".[7] In this application, the "measurements of the agent's behavior" would consist of observations of human preferences, and the "reward function being optimized" would be the human's posited utility function. In these terms, the problem of IRL could be given a solution by the proof of the vNM theorem if the human were ideally rational. Ng and Russell do not make the assumption of ideal rationality and instead use a statistical model to obtain an optimal reward function.

§3.1 Ng and Russell's Markov Decision Process

Ng and Russell's solution to IRL is to model the agent's behavior as a Markov Decision Process (MDP). In an MDP, the agent's behavior is modeled as a state machine that transitions between states based on the agent's actions. The agent's actions are determined by a policy, which is a mapping from states to actions. The agent's reward function is a function of the state and action, and the agent's goal is to maximize the expected sum of rewards over time.

[6] Ng and Russell.
[7] Ng and Russell, Introduction.
[8] Ng and Russell, Section 2.1.

§3.1.1 The MDP Formalism

In their formalism[8] for a finite space of states, the MDP consists of a 5-tuple $(S, A, \{P_{s\alpha}, \gamma, R)$, where:

- S is the finite set of N **states**
- A is the finite set of k **actions**
- $P_{s\alpha}(\cdot)$ is the **transition probability** of taking action $\alpha \in A$ in state $s \in S$
- $\gamma \in [0, 1]$ is the **discount factor**
- $R : S \to \mathbb{R}$ is the **reinforcement function**, also sometimes called the **reward function**

Having defined the MDP, they define a **policy function** as any function $\pi : S \to A$, and the **value function** for some policy function π as $V^\pi(s_1) = \mathbb{E}\left[\sum_i^\infty \gamma^{i-1} R(s_i) \mid \pi\right]$. This expectation is over all trajectories $s_1, s_2, s_3, \dots$ which start in state s_1, follow the policy $\pi(s_i)$ to determine the action α_i, and transition to s_{i+1} with probability $P_{s_i \alpha_i}(s_{i+1})$. Intuitively, $V^\pi(s_1)$ is the expected sum of time-discounted rewards (obtained from the reinforcement function) from starting in state s_1 and following policy π forever.

On this formalism, the goal of standard (non-inverse) reinforcement learning is to find π which maximizes $V^\pi(s)$ over all $s \in S$. By contrast, the goal of inverse reinforcement learning is to find the families of reward functions $\boldsymbol{R}$ such that, given a partially specified MDP $(S, A, P_{s\alpha}, \gamma, \cdot)$ and a given policy π, π maximizes $V^\pi(s)$ for the fully specified MDP $(S, A, P_{s\alpha}, \gamma, R \in \boldsymbol{R})$. In other words, given a combination of states, actions, transition probabilities, discount factor, and policy, we must reconstruct the reward function(s) that makes the given policy optimal.

§3.1.2 The Linear Program Solution to the MDP

Ng and Russell present a linear program solution to the MDP they formulate above. To simplify their notation, they use a relabeling trick[9] in which the labels for actions and the probabilities of transitions are systematically permuted such that the optimal policy $\pi(s) = \alpha_1$ for all $s \in S$. Ng and Russell show in their **Theorem 3** that this π is optimal iff

$$\forall_{i \in (2 \dots k)} \left(\boldsymbol{P}_{\{\alpha_1\}} - \boldsymbol{P}_{\{\alpha_i\}}\right)\left(\boldsymbol{I} - \gamma \boldsymbol{P}_{\{\alpha_1\}}\right)^{-1} \boldsymbol{R} \geq \boldsymbol{0}$$

Here, $\boldsymbol{P}_{\{\alpha_x\}}$ is the probability transition matrix whose (i, j)th entry is $P_{s_i \alpha_x}(s_j)$ for $s_i, s_j \in S$ and $\alpha_x \in A$, or in other words, the probability of transitioning from state s_i to state s_j after taking action α_x. This imposes a set of linear constraints on the possible reward Functions $\boldsymbol{R}$. To obtain a unique solution, Ng and Russell seek to maximize the

[9] I will not discuss this trick for reasons of space

penalty, according to R, of deviating from the optimal policy π. This yields the following linear program:

$$\boxed{\begin{aligned}
&\text{maximize } \left(\sum_{i=1}^{N} \min_{\alpha\in A\backslash\alpha_1} \left\{\left(\boldsymbol{P}_{\{\alpha_1\}}(i) - \boldsymbol{P}_{\{\alpha_i\}}(i)\right)\left(\boldsymbol{I} - \gamma \boldsymbol{P}_{\{\alpha_1\}}\right)^{-1}\boldsymbol{R}\right\}\right) - \lambda \left\|\boldsymbol{R}\right\|_1 \\
&\qquad \text{s.t. } \forall_{\alpha\in A\backslash\alpha_1}\left(\boldsymbol{P}_{\{\alpha_1\}} - \boldsymbol{P}_{\{\alpha\}}\right)\left(\boldsymbol{I} - \gamma \boldsymbol{P}_{\{\alpha_1\}}\right)^{-1}\boldsymbol{R} \underset{v}{\succeq} \mathbf{0}) \\
&\qquad\quad \forall_{i\in 1\ldots N}\ |\boldsymbol{R}_i| \underset{v}{\preceq} R_{\max}
\end{aligned}}$$

where $\underset{v}{\preceq}$ and $\underset{v}{\succeq}$ are the conjunction of element-wise $\leq$ and $\geq$ relations between corresponding elements of vectors. Note that the function to be optimized is linear in $\boldsymbol{R}$.

§3.2 Using the MDP to Reconstruct Utility from Preferences

I will now propose the most natural way in which the problem of the vNM theorem—i.e., to reconstruct a utility function from elicited preferences—can be addressed using the formalism of the MDP. This will illustrate key limitations of the MDP formulation at the heart of IRL, showing that a more sophisticated model that explicitly models error and preference change is needed.

First, note that Ng and Russell's MDP formalism assigns rewards to states, not to actions or choices. The closest parallel in the vNM presentation of the problem is the utility function, which assigns a utility to each lottery. Furthermore, the MDP formalism assigns a constant utility to each state, since R is time-invariant. Similarly, the vNM theorem assumes that the utility function is constant over time. Hence, the reward function in the MDP formalism is a clear parallel to the utility function in the vNM theorem.

In addition, the problem of IRL in the MDP is to find the reward functions whose time-discounted sum over time is maximized by the given policy. Similarly, the vNM theorem shows how to construct a utility function which is *at each time* maximized by the given preferences. Since the value of the utility function at some time is only dependent on the lottery, not the value of the utility function at any other time, we can express the goal of the vNM theorem as finding a utility function which maximizes the (non-time-discounted) sum of utilities at each time. This suggests setting $\gamma = 1$ in the MDP formalism.

The functional analogue of the policy function $\pi : S \to A$ in the MDP is the set of observed preference $A_i \succ B_i$ in the vNM theorem. The vNM theorem gives agents only three options at each time step: given two lotteries A and B, either choose A over B, choose B over A, or be indifferent between A and B. In the MDP formalism, all actions

$\alpha \in A$ are possible at any time, in any state. Thus, the most faithful representation of the vNM problem in the MDP's terms constrains A to a set of three actions. Since the action cannot represent the information of which lotteries are presented, we must instead represent that information in the MDP's state. We can do so using the following construction. Let $\mathcal{L}$ be the set of lotteries that can be presented to the agent in the vNM problem, and let $\underset{\mathcal{L}}{<}$ be any strict linear order on $\mathcal{L}$. Then, the MDP state space S can be defined as follows:

$$S = \left\{(x, i, j) \mid x, i, j \in \mathcal{L}; i \underset{\mathcal{L}}{<} j\right\}$$

State $s = (x, i, j) \in S$ represents the state in which the agent has just chosen x and is about to choose between i and j. The actions available are:

$$A = \{\text{left}, \text{right}, \text{indifferent}\}$$

Here, choosing "left" means choosing i over j, choosing "right" means choosing j over i, and choosing "indifferent" means being indifferent between i and j. Thus, choosing "left" in state $s = (x, i, j)$ transitions one to some state $s' = (i, \cdot, \cdot)$. Accordingly, the observation that $\pi(s) = i$ can be translated into vNM terms as follows:

$$\pi((x, i, j)) = \begin{cases} \text{left} & \rightarrow i \succ j \rightarrow U_i > U_j \\ \text{right} & \rightarrow j \succ i \rightarrow U_j > U_i \\ \text{indifferent} & \rightarrow i \sim j \rightarrow U_i = U_j \end{cases}$$

In the vNM setup, the agent gains utility U_i, regardless of what their next choice will be between. Thus, in the MDP, following this transition, the agent's value function should depend only on the first element of the 3-tuple forming the state.

To represent this feature of the state, we can add the constraint that

$$R(i, \cdot, \cdot) = \text{fn}(i)$$

This constraint can be added to the linear program in Section 3.1.2 as a constraint on the elements of $\boldsymbol{R}$ when represented as a vector.

Since, in the vNM formulation, the agent gains some utility for certain upon making a certain choice, we can also give the following form for $P_{s\alpha}(\cdot)$, where $c = \frac{(|S|)(|S| -1)}{2}$ is the number of unique choices the agent could face:

$$P_{s\alpha}(s'; s = (x, i, j), s' = (y, k, l)) = \begin{cases} \alpha = \text{left} : & \begin{cases} 0 \text{ if } i{\neq}y \\ \frac{1}{c} \text{ if } i{=}y \end{cases} \\ \alpha = \text{right} : & \begin{cases} 0 \text{ if } j{\neq}y \\ \frac{1}{c} \text{ if } j{=}y \end{cases} \\ \alpha = \text{indifferent} : & \begin{cases} 0 & \text{if } i{\neq}y \text{ and } j{\neq}y \\ \frac{1}{2c} & \text{if } i{=}y \text{ or } j{=}y \end{cases} \end{cases}$$

In other words, the probability of transitioning into a state (y, k, l) is zero if the agent's previous state (x, i, j) combined with their choice (left, right, indifferent) is inconsistent with the new state of the form $(y, \cdot, \cdot)$. Otherwise, the probability mass is uniformly distributed over all states of the form $(y, \cdot, \cdot)$, since the next choice the agent faces is independent of the previous choice.

Thus, the IRL MDP $(S, A, P_{s\alpha}, \gamma, \cdot)$ and policy π that corresponds to the problem faced in the vNM theorem can be described as follows:

$$\begin{aligned}
S &= \left\{(x, i, j) \mid x, i, j \in \mathcal{L}; i \underset{\mathcal{L}}{<} j\right\} \\
A &= \{\text{left}, \text{right}, \text{indifferent}\} \\
P_{s\alpha}(\cdot) &= \begin{cases} \alpha = \text{left} : & \begin{cases} 0 \text{ if } i{\neq}y \\ \frac{1}{c} \text{ if } i{=}y \end{cases} \\ \alpha = \text{right} : & \begin{cases} 0 \text{ if } j{\neq}y \\ \frac{1}{c} \text{ if } j{=}y \end{cases} \\ \alpha = \text{indifferent} : & \begin{cases} 0 \quad \text{if } i{\neq}y \text{ and } j{\neq}y \\ \frac{1}{2c} \text{ if } i{=}y \text{ or } j{=}y \end{cases} \end{cases} \\
\gamma &= 1 \\
R(i, \cdot, \cdot) &= \text{fn}(i) \\
\pi((\cdot, i, j)) &= \begin{cases} \text{left} & \text{if } i \succ j \\ \text{right} & \text{if } j \succ i \\ \text{indifferent} & \text{if } i \sim j \end{cases}
\end{aligned}$$

Having constructed the MDP that results from using inverse reinforcement learning to solve the problem considered in the vNM theorem, I will now argue that this model does not have the theoretical resources to model agents that commit errors in predictable ways and exhibit preference change.

§3.3 Limitations of the MDP

In Section 2.2 and Section 2.3, I argued that we can model humans as agents which commit errors in predictable ways and whose utility functions change in predictable ways time. In particular, I argued that the probability of error $\varepsilon(A, B)$ between two lotteries A and B is a function of the utilities of A and B, and that change in the utility function can be modeled as a Lévy flight, a special kind of random walk. Examination of the MDP formalism shows that it can accommodate neither of these claims about human behavior. I will justify these claims in this order, first considering error and then preference change.

§3.3.1 Error in the MDP

In formulating the optimization problem of IRL, Ng and Russell assume that π is the optimal policy. Here, optimality is defined as the condition that π is such that $V^{\pi}(s)$ is maximized for all $s \in S$. Furthermore, as argued above, π is determined from the observed preferences of the agent: we stipulate that $\pi((\cdot, i, j)) = \text{left}$ iff the the behavioral preferences include i being chosen over j. This necessity is similar to the one present in the vNM theorem, in which $\alpha \succ b$ conceptually guarantees that $U_\alpha > U_b$.

Suppose that the agent commits an error: their true utility function is such that $U_\alpha < U_b$ but the agent chooses α over b. Then, their policy function π will be such that $\pi((\cdot, \alpha, b)) = \text{left}$. Given the structure of $P_{s\alpha}(\cdot)$, we know that the subsequent state will be of the form $(\alpha, \cdot, \cdot)$, and the agent will obtain reward $R(\alpha, \cdot, \cdot)$. By contrast, if the agent had instead (correctly) chosen b over α, the subsequent state would be of form $(b, \cdot, \cdot)$, and the agent would obtain reward $R(b, \cdot, \cdot)$. Since we are using $R(x, \cdot, \cdot)$ to represent the utility of the lottery x, we must have $R(\alpha, \cdot, \cdot) < R(b, \cdot, \cdot)$.

However, these conditions violate the Bellman Optimality theorem used by Ng and Russell to formulate their linear program. The Bellman Optimality theorem states:

$$\begin{aligned} \pi(s) &\in \arg\max_{\alpha \in A} Q^\pi(s, \alpha) \\ &= \arg\max_{\alpha \in A} \left\{ R(s) + \gamma \sum_{s' \in S} P_{s\alpha}(s') V^\pi(s') \right\} \end{aligned}$$

Plugging in our 3-tuple structure for states and setting $\gamma = 1$, we instead have:

$$\begin{aligned} \pi((x, i, j)) &\in \arg\max\nolimits_{\alpha \in A} Q^\pi((x, i, j), \alpha) \\ &= \arg\max\nolimits_{\alpha \in A} \Big\{ R(x, \cdot, \cdot) + \textstyle\sum_{(y, i', j') \in S} P_{(x,i,j)\alpha}((y, i', j')) V^\pi((y, i', j')) \Big\} \\ &= \arg\max\nolimits_{\alpha \in A} \Big\{ \textstyle\sum_{(y, i', j') \in S} P_{(x,i,j)\alpha}((y, i', j')) V^\pi((y, i', j')) \Big\} \end{aligned}$$

Thus, the optimal action at state (x, i, j) is the one which maximizes the expected value of the value function V^π weighted by the probability of each subsequent state. However, noting that the reward function R is a function only of the first term of the state (representing the chosen lottery) and that all choices betwen subsequent lotteries are equiprobable independent of the currently chosen lottery, we can see that this expectation depends only on $R(y, \cdot, \cdot)$ where y is the lottery chosen from (x, i, j) by the action, namely i if $\alpha = \text{left}$ and j if $\alpha = \text{right}$. Thus,

$$\pi((x, i, j)) = \arg\max_{\alpha \in A} \{ R(y, \cdot, \cdot) \mid y = i \text{ if } \alpha = \text{left}, y = j \text{ if } \alpha = \text{right} \}$$

Hence,

$$\pi((x, a, b)) = \text{left} \longleftrightarrow R(a, \cdot, \cdot) > R(b, \cdot, \cdot)$$

But this conclusion is simply the denial of the possibility that an agent can make a mistake. Hence, the MDP formulated above cannot account for the possibility of error.

§3.3.2 Preference Change in the MDP

In the MDP, the reward function R is time-invariant: it always assigns the same reward to each state. In turn, this reward is added to the value function V^π with a time-discount factor of γ^t if it occurs t time steps in the future. I argued above that $\gamma = 1$, so the MDP formulation of the vNM problem does not time-discount. Even if we let $0 < \gamma < 1$, however, the MDP formalism cannot account for all forms of preference change. A basic case of preference change involves changing one's mind about which of A or B to choose: $A \underset{t_1}{\succ} B$ but $B \underset{t_2}{\succ} A$. In this case, the random-walk model of preference change would time-index the utilities of A and B, such that $U_A(t_1) > U_B(t_1)$ but $U_B(t_2) > U_A(t_2)$. However, the MDP formalism's reward function R is a function only of the state, not the time. Thus, if $R(s_1, \cdot, \cdot) > R(s_2, \cdot, \cdot)$, then this inequality will hold at all times. Even if we introduce a time-discount $0 < \gamma < 1$,

$$R(s_1, \cdot, \cdot) > R(s_2, \cdot, \cdot) \to \gamma^t R(s_1, \cdot, \cdot) > \gamma^t R(s_2, \cdot, \cdot)$$

for all t. Introducing negative γ would allow $R(s_1, \cdot, \cdot) > R(s_2, \cdot, \cdot)$ but $\gamma^t R(s_1, \cdot, \cdot) < \gamma^t R(s_2, \cdot, \cdot)$ for some t. However, this is not a plausible model of preference change, since whether the lottery obtained in s_1 is preferred to the lottery obtained in s_2 would depend entirely on whether an even or odd number of time steps have passed.

Another possible way to model preference change is to encode changes in the utility of a lottery within the state. In the MDP I formulated above, I used:

$$S = \left\{(x, i, j) \mid x, i, j \in \mathcal{L}; i \underset{\mathcal{L}}{<} j\right\}$$

where x represents the lottery that the agent has just chosen. In turn, the reward function depends on x. To account for change in the utility of x over time, we could instead use a state of the form

$$S = \left\{(x, i, j; t) \mid x, i, j \in \mathcal{L}; i \underset{\mathcal{L}}{<} j; t \in \mathbb{N}\right\}$$

With this addition, the reward function could be sensitive to both x and t, which would allow $R(x_1, \cdot, \cdot, t_1) > R(x_2, \cdot, \cdot, t_2)$ while $R(x_1, \cdot, \cdot, t_2) < R(x_2, \cdot, \cdot, t_2)$.

Although this change would allow for time-variant preferences, restrictions within the MDP formalism would not allow modeling preference change as a statistcally governed process. As described in Section 2.3, preference change can be expected to typically be small but contain occasional large jumps. Constraints of this form can be expressed using a probability distribution for $R(x, \cdot, \cdot, t_j)$ conditional on $R(x, \cdot, \cdot, t_i)$ for $i < j$. To maximally satisfy these constraints, we would need to maximize the joint likelihood produced by the products of $\Pr\left(R(x, \cdot, \cdot, t_j) \mid R(x, \cdot, \cdot, t_i)\right)$.

In the linear program that Ng and Russell give for the MDP (Section 3.1.2), the function to be optimized is the sum of the counterfactual penalties from deviating from the optimal policy, which was a linear function of the reward function. By contrast, the joint likelihood that needs to be optimized, if we are to impose constraints on the evolution of the reward function over time, is a product:

$$\text{maximize} \prod_{x\in\mathcal{L},1\le t_i<t_j} \Pr\big(R\big(x,\cdot,\cdot,t_j\big) \mid R(x,\cdot,\cdot,t_i)\big)$$

which is equivalent to the following optimization constraint due to monotonicity of log:

$$\text{maximize} \sum_{x\in\mathcal{L},1\le t_i<t_j} \log\Pr\big(R\big(x,\cdot,\cdot,t_j\big) \mid R(x,\cdot,\cdot,t_i)\big)$$

This sum will only be a linear function of the elements of R if $\log\big(\Pr\big(R\big(x,\cdot,\cdot,t_j\big) \mid R(x,\cdot,\cdot,t_i)\big)\big)$ is a linear function of $R(x,\cdot,\cdot,t_i)$ and $R\big(x,\cdot,\cdot,t_j\big)$. This constrains us to conditional probabilities of the form

$$\begin{aligned}\Pr\big(R\big(x,\cdot,\cdot,t_j\big) \mid R(x,\cdot,\cdot,t_i)\big) &= \exp\big(a + bR(x,\cdot,\cdot,t_i) + cR\big(x,\cdot,\cdot,t_j\big)\big)\\ \log\Pr\big(R\big(x,\cdot,\cdot,t_j\big) \mid R(x,\cdot,\cdot,t_i)\big) &= a + bR(x,\cdot,\cdot,t_i) + cR\big(x,\cdot,\cdot,t_j\big)\end{aligned}$$

This would make each term within the maximized sum a linear combination of $R(x,\cdot,\cdot,t_i)$ and $R\big(x,\cdot,\cdot,t_j\big)$. In addition, note that a probability function of the form described above would not be a valid probability function over infinite support, since its integral from $-\infty$ to ∞ always diverges.

Although some model of preference change could be produced according to these constraints, the constraint that the conditional probability is a linear function of the reward function is highly inflexible. A better model of preference change would allow for a broad range of possible conditional probability functions for the utility at a later time conditional on the utility at a previous time.

§3.4 Moving beyond the MDP and IRL

As we have seen in this section, Ng and Russell's formulation of IRL as a MDP is not flexible enough to model the possibility of error or preference change. Like the vNM theorem's proof, the MDP formalism makes the connection too strong between policy or behavior on one hand, and the utility or reward on the other. Furthermore, although it may be possible to model some sort of preference change in the MDP by injecting time into the state, this only allows a very specific and psychologically implausible model for preference change. This latter constraint arises from the linear-programming approach that Ng and Russell take to IRL, which requires that the reward function be a linear function of the state.

To provide a more flexible model of preference change, I will now propose a Bayesian model of an agent's utility function given their preferences which explicitly accounts for error and preference change.

§4 General models for errors and changing utility functions

In this section, I will explore various Bayesian models that account for errors and changes in utility functions over time. I will begin by discussing the perfectly rational case, where preferences align perfectly with a constructed utility function. Next, I will introduce the predictable error case, which incorporates the likelihood of errors in decision-making. Finally, I will examine scenarios where utility functions evolve over time, highlighting the implications of these changes on decision-making processes and the statistical modeling of preferences. Here, I leave the models of error and preference change highly general so that a variety of models can be explored. Section 5 then introduces specific functions for error and preference change motivated by the psychological observations in Section 2.

§4.1 Perfectly Rational Case

The proof of the vNM theorem is constructive: it constructs a utility function U that represents a set of preferences that obey the vNM axioms. If we use this construction not as a mathematical proof but rather as an algorithm, we can thus use demonstrated or elicited preferences as evidence that we use to model an agent's utility function. More concretely, if we know an agent's preferences over some lotteries A, B, C, ... we can construct a model the agent's utility function over A, B, C, and other lotteries.

In the case of the ideally rational agent, we take the fact that A is chosen over B to guarantee that $U_A > U_B$ and vice versa. This is because we assume that the agent does not make mistakes, nor does their utility function ever change. This is a matter of conceptual necessity. However, if forced to express this relationship in Bayesian terms, we might write:

$$p(U_A > U_B \mid A \succ B) = 1$$

That this conditional probability is 1 represents that the probability that a preference does not correctly represent the utility function is 0.

§4.2 Predictable Error Case

By contrast, if we want to include the possibility of error, we might instead write:

$$p(U_A > U_B \mid A \succ B) = 1 - \varepsilon(A, B)$$

where ε is a probability measure over ordered pairs of lotteries. For simplicity's sake I assume here that ε can be equivalently expressed in terms of A and B directly, or over UA and UB, since the only relevant features of A and B are their utilities. Of course, this is not psychologically plausible, since even if I am indifferent between lotteries A and B (say, $1000 for sure and a 10% chance of $10000, or 10% chance of $11000 if I am slightly riskaverse), I may be prone to error in different ways given the presentation of the lotteries independent of the utilities.

§4.3 Changing Utility Functions Without Error

Now, consider the case where the utility function changes over time. Let $U_X(t)$ be the utility of lottery X at time t. If the utility function can be statistically modeled, then we can express a probability distribution for $U_{X(t)}$ conditional on the past value of $U_X(t-1)$. In the constant case, we would have (replacing U_X with U for simplicity's sake):

$$\Pr(U(t) = U(t-1)) = f_{U(t)|U(t-1)}(u|v) = \mathbf{1}_{u=v}$$

where $\mathbf{1}_{u=v}$ is the indicator function that is 1 if $u = v$ and 0 otherwise. This represents the fact that the previous utility function completely determines the current utility function.

We can also represent this in a different form: Let $\Delta_t = U(t) - U(t-1)$. Then $\Delta_t = 0$. By contrast, when utilities change, Δ_t will also be non-constant. Then, we have the following conditional probability density function for $U(t)$:

$$f_{\{U(t)\ |\ U(t-1)\}}(u|v) = f_{\Delta_t}(u-v)$$

where f_{Δ_t} is the probability density function for Δ_t. If all Δ_t are i.i.d., then we can replace Δ_t with simply Δ and use f_Δ instead of f_{Δ_t}, so

$$f_{\{U(t)\ |\ U(t-1)\}}(u|v) = f_\Delta(u-v)$$

and

$$f_{U(t),U(t-1)}(u,v) = f_{U(t)\ |\ U(t-1)}(u|v) f_{U(t-1)}(v) = f_\Delta(u-v) f_{U_{t-1}}(v)$$

We can also marginalize over $U(t-1)$ to get the PDF for $U(t)$, where S is the support of $U(\cdot)$:

$$f_{U(t)}(u) = \int_S f_\Delta(u-v) f_{U(t-1)}(v) dv$$

However, we do not (only) learn about the current utility function directly from the previous utility function. Rather, we gain information at each time using elicited preferences. Thus, our goal is to obtain a conditional probability density for $U(t)$ that depends only on the elicited preferences.

I will make the simplifying assumption that rather than comparing two lotteries, we can directly determine the agent's preferences between some lottery X and a sure outcome of constant known utility n. Most simply, we could offer the agent a choice between a sure outcome of X or a sure amount of utility n, thus effectively 'bidding' a purchase price of utility n for the agent to purchase the lottery. Alternatively, we might assume we have available a constant outcome W with a constant utility of 1 and a constant outcome L with a constant utility of L. Then, we can attempt to determine the utility of an outcome X by comparing lotteries of X vs. a sure outcome of W if $U_X(t) > 1$ and comparing lotteries of W vs. a sure outcome of X if $0 < U_X(t) < 1$. If X has negative utility, we can compare against L instead of W. Both would provide an operationalized version of evidence that $U_X(t) > n$ or $U_X(t) < n$ for any real n.

These are admittedly unrealistic assumptions, but they are a starting point for the purposes of this paper. For the sake of this paper, I will assume the first 'bidding' strategy is available and that we can directly compare any lottery X against a sure outcome of utility n.

The conditional probability density for $U_X(t)$ conditional only on the most recent elicited preference can thus be derived as follows:

$$f_{U(t),U(t-1)\ |\ U(t)>n}(u,v) = \frac{f_{U(t),U(t-1)}(u,v)\mathbf{1}_{u>n}}{\Pr(U(t)>n)} = \frac{\mathbf{1}_{u>n}}{\Pr(U(t)>n)} f_\Delta(u-v) f_{U(t-1)}(v)$$

$$f_{U(t),U(t-1)\ |\ U(t)<n}(u,v) = \frac{f_{U(t),U(t-1)}(u,v)\mathbf{1}_{u<n}}{\Pr(U(t)<n)} = \frac{\mathbf{1}_{u<n}}{\Pr(U(t)<n)} f_\Delta(u-v) f_{U(t-1)}(v)$$

Marginalizing out $U(t-1)$:

$$f_{U(t)\ |\ U(t)>n}(u) = \frac{\mathbf{1}_{u>n}}{\Pr(U(t)>n)} \int_S f_{\Delta(u-v)} f_{U(t-1)}(v)dv$$
$$f_{U(t)\ |\ U(t)<n}(u) = \frac{\mathbf{1}_{u<n}}{\Pr(U(t)<n)} \int_S f_{\Delta(u-v)} f_{U(t-1)}(v)dv$$

This gives an iterative method for computing the new posterior for $U(t)$ after learning if $U(t) > n$ or $U(t) < n$. Although this integral is likely analytically intractable, it can be numerically computed using a grid method or a particle filter. The new posterior is whichever of $f_{U(t)\ |\ U(t)>n}(u)$ and $f_{U(t)\ |\ U(t)<n}(u)$ corresponds to the observed preference.

§4.4 Changing Utility Functions With Error

Finally, let us consider the case where utilities change and there is the possibility of error. Here, we cannot simply infer from an exhibited preference for lottery X over a sure utility of n that $U_X(t) > n$.

Let Υ_n be a sure outcome of utility n. Then, at each time we learn either the preference $X \succ_t \Upsilon_n$ or $X \prec_t \Upsilon_n$. We can give the probability of each event in terms of the probability of error as follows:

$$\Pr(X \succ_t \Upsilon_n) = \begin{cases} 1-\varepsilon(X,\Upsilon_n) & \text{if } U_X(t) > n \\ \varepsilon(X,\Upsilon_n) & \text{if } U_X(t) < n \end{cases}$$

$$\Pr(X \prec_t \Upsilon_n) = \begin{cases} \varepsilon(X,\Upsilon_n) & \text{if } U_X(t) > n \\ 1-\varepsilon(X,\Upsilon_n) & \text{if } U_X(t) < n \end{cases}$$

We can simplify away the casework by switching terminology from an error function $\varepsilon(A, B)$ to a policy function $\pi(U_A, U_B)$. Recall that $\varepsilon(A, B)$ is the probability that the wrong preference is indicated compared to the true utilities; that is, it is the probability that A is chosen over B in the case that $U_A < U_B$, or the probability that B is chosen over A in the case that $U_B < U_A$. Although it is natural to think of the relationship between choice and utility in terms of an error or mismatch, it is mathematically simpler to express the probability of choice solely in terms of the utilities, so

$$\Pr(X \succ_t \Upsilon_n) = \pi(U_X(t), n)$$
$$\pi(U_A, U_B) = \begin{cases} 1 - \varepsilon(A, B) & \text{if } U_A > U_B \\ \varepsilon(A, B) & \text{otherwise} \end{cases}$$

This imposes the property on π that $\pi(a, b) = 1 - \pi(b, a)$. Replacing the indicator function 1 for the events that $u > n$ and $u < n$ above, we have:

$$\begin{aligned} f_{U(t),U(t-1)\mid X \succ_t \Upsilon_n}(u, v|w) &= \frac{f_{U(t),U(t-1)}(u, v)\pi(u, w)}{\int_S f_{U(t),U(t-1)}(u', v)\pi(u', w)du'} \\ &= \frac{f_\Delta(u - v)f_{U(t-1)}(v)\pi(u, w)}{\int_S f_\Delta(u' - v)f_{U(t-1)}(v)\pi(u', w)du'} \end{aligned}$$

$$\begin{aligned} f_{U(t),U(t-1)\mid X \prec_t \Upsilon_n}(u, v|w) &= \frac{f_{U(t),U(t-1)}(u, v)f_{U(t-1)}(v)\pi(w, u)}{\int_S f_{U(t),U(t-1)}(u', v)f_{U(t-1)}(v)\pi(w, u')du'} \\ &= \frac{f_\Delta(u - v)f_{U(t-1)}(v)\pi(w, u)}{\int_S f_\Delta(u' - v)f_{U(t-1)}(v)\pi(w, u')du'} \end{aligned}$$

Again, marginalizing out $U(t - 1)$ gives us the new posterior for $U(t)$:

$$\begin{aligned} f_{U(t)\mid X \succ_t \Upsilon_n}(u|w) &= \int_S \frac{f_\Delta(u - v')f_{U(t-1)}(v')\pi(u, w)}{\int_S f_\Delta(u' - v')f_{U(t-1)}(v')\pi(u', w)du'} dv' \\ &= \pi(u, w) \int_S \frac{f_\Delta(u - v')f_{U(t-1)}(v')}{\int_S f_\Delta(u' - v')f_{U(t-1)}(v')\pi(u', w)du'} dv' \end{aligned}$$

$$\begin{aligned} f_{U(t)\mid X \prec_t \Upsilon_n}(u|w) &= \int_S \frac{f_\Delta(u - v')f_{U(t-1)}(v')\pi(w, u)}{\int_S f_\Delta(u' - v')f_{U(t-1)}(v')\pi(w, u')du'} dv' \\ &= \pi(w, u) \int_S \frac{f_\Delta(u - v')f_{U(t-1)}(v')}{\int_S f_\Delta(u' - v')f_{U(t-1)}(v')\pi(u', w)du'} dv' \end{aligned}$$

This again gives an iterative method for computing the new posterior for $U(t)$ given the posterior of $U(t - 1)$ and the observation of either $X \succ_t \Upsilon_n$ or $X \prec_t \Upsilon_n$. As above, this requires an analytically intractable computation, but one which can be numerically approximated.

In the following section, I will provide specific error and utility-change functions that adhere to the psychological models of error and utility-change proposed in Section 1. By using these specific functions, we can derive the specific form for the posterior for $U(t)$ given the posterior of $U(t-1)$ and the observation of either $X \succ_t \Upsilon_n$ or $X \prec_t \Upsilon_n$.

§5 Specific error policy and utility-change functions

§5.1 Sigmoid Error Function and Policy

In the original formulation for a predictable error, we had

$$\varepsilon(A, B) = f(U_A, U_B)$$

where f is some function of U_A and U_B. Later, we replaced ε with π, which returns the probability of either $A \succ B$ or $B \succ A$ independent of the relative values of U_A and U_B. This required the additional constraint of $\pi(a, b) = 1 - \pi(b, a)$.

Hadfield-Mennell (2016) uses a sigmoid function as an error policy. Given two lotteries with utilties a and b, the probability of choosing the lottery with utility a is given by

$$\pi(a, b) = \frac{1}{1 + e^{\frac{b-a}{\sigma_\varepsilon}}}$$

where σ_ε is a parameter that controls the steepness of the sigmoid.[10] The smaller the value of $b - a$ — i.e., the more choiceworthy the lottery with utility a is — the more likely the agent is to choose the lottery of utility a, as the denominator of the sigmoid remains close to 1. By the same token, as $b - a$ increases, the denominator of the sigmoid increases, and the probability of choosing the lottery of utility a decreases. When a and b are close, then $\pi(a, b)$ approaches 0.5. As σ_ε increases, a given difference $|b - a|$ will produce a less decisively correct agent than for a smaller value of σ. Concretely, when $|b - a| = \sigma_\varepsilon$, then there is roughly a 27% chance of error.

If we include this policy function in the general form for the posterior, we have:

$$\begin{aligned} f_{U(t) \mid X \succ_t \text{«} \Upsilon_n}(u|w; \sigma_\varepsilon) &= \pi(u, w) \int_S \frac{f_\Delta(u - v') f_{U(t-1)}(v')}{\int_S f_\Delta(u' - v') f_{U(t-1)}(v') \pi(u', w) du'} dv' \\ &= \left(1 + e^{\frac{w-u}{\sigma_\varepsilon}}\right)^{-1} \int_S \frac{f_\Delta(u - v') f_{U(t-1)}(v')}{\int_S f_\Delta(u' - v') f_{U(t-1)}(v') \left(1 + e^{\frac{w-u'}{\sigma_\varepsilon}}\right)^{-1} du'} dv' \end{aligned}$$

$$\begin{aligned} f_{U(t) \mid X \prec_t \Upsilon_n}(u|w; \sigma_\varepsilon) &= \pi(w, u) \int_S \frac{f_\Delta(u - v') f_{U(t-1)}(v')}{\int_S f_\Delta(u' - v') f_{U(t-1)}(v') \pi(w, u') du'} dv' \\ &= \left(1 + e^{\frac{u-w}{\sigma_\varepsilon}}\right)^{-1} \int_S \frac{f_\Delta(u - v') f_{U(t-1)}(v')}{\int_S f_\Delta(u' - v') f_{U(t-1)}(v') \left(1 + e^{\frac{u'-w}{\sigma_\varepsilon}}\right)^{-1} du'} dv' \end{aligned}$$

This function contains one further general element: f_Δ.

[10] Hadfield-Menell et al., "The Off-Switch Game".

§5.2 Gaussian Step Size Utility-Change Function

One natural candidate for f_Δ is a Gaussian distribution with mean 0 and standard deviation σ. This is the simplest case of a stable utility-change function that is non-constant, though it does not satisfy the definition of a Lévy flight, since the tails of the distribution are not heavy. If f_Δ is a Gaussian distribution with mean 0 and standard deviation σ, then we have:

$$\delta \sim \mathcal{N}\big(0, \sigma_\Delta^2\big)$$
$$f_\Delta(\delta) = \frac{1}{\sqrt{2\pi\sigma_\Delta^2}} e^{-\frac{\delta^2}{2\sigma_\Delta^2}}$$

For reasons of space, and since I will not use this model, I will not write the complete posterior for $U(t)$ using the Gaussian step size function.

§5.3 Cauchy Step Size Utility-Change Function

As described in Section 2, the Cauchy distribution is a natural choice for this application, since it satisfies the stability and heavy-tailed requirements of a Lévy flight while also having support over $\mathbb{R}$. If f_Δ is a Cauchy distribution with principal value and median[11] 0 and scale parameter γ, then we have:

$$\delta \sim \text{Cauchy}(0, \gamma)$$
$$f_\Delta(\delta) = \frac{\gamma}{\pi(\gamma^2 + \delta^2)}$$

Plugging this form for f_Δ into the posterior of $U(t)$ above:

$$\begin{aligned}
f_{U(t)\,|\,X \succ_t \Upsilon_n}(u|w; \sigma_\varepsilon, \gamma) &= \Big(1 + e^{\frac{w-u}{\sigma_\varepsilon}}\Big)^{-1} \int_S \frac{f_\Delta(u - v') f_{U(t-1)}(v')}{\int_S f_\Delta(u' - v') f_{U(t-1)}(v') \Big(1 + e^{\frac{w-u'}{\sigma_\varepsilon}}\Big)^{-1} du'} dv' \\
&= \Big(1 + e^{\frac{w-u}{\sigma_\varepsilon}}\Big)^{-1} \int_S \frac{\frac{\gamma}{\pi\left(\gamma^2 + (u-v')^2\right)} f_{U(t-1)}(v')}{\int_S \frac{\gamma}{\left(\gamma^2 + (u'-v')^2\right)} f_{U(t-1)}(v') \Big(1 + e^{\frac{w-u'}{\sigma_\varepsilon}}\Big)^{-1} du'} dv' \\
&= \Big(1 + e^{\frac{w-u}{\sigma_\varepsilon}}\Big)^{-1} \int_S \frac{\frac{1}{\left(\gamma^2 + (u-v')^2\right)} f_{U(t-1)}(v')}{\int_S \frac{1}{\left(\gamma^2 + (u'-v')^2\right)} f_{U(t-1)}(v') \Big(1 + e^{\frac{w-u'}{\sigma_\varepsilon}}\Big)^{-1} du'} dv'
\end{aligned}$$

[11] The Cauchy distribution has a well-defined median and a mean over a finite restricted interval $[-T, T]$, but a mean of undefined over its whole support. For the purposes of this paper, I will use the median as the measure of central tendency.

$$f_{U(t)\ \mid\ X \prec_t \Upsilon_n}(u|w;\sigma_\varepsilon,\gamma) = \left(1+e^{\frac{u-w}{\sigma_\varepsilon}}\right)^{-1} \int_S \frac{\frac{1}{\left(\gamma^2+(u-v')^2\right)} f_{U(t-1)}(v')}{\int_S \frac{1}{\left(\gamma^2+(u'-v')^2\right)} f_{U(t-1)}(v') \left(1+e^{\frac{u'-w}{\sigma_\varepsilon}}\right)^{-1} du'} dv'$$

Although this is clearly analytically intractable, it can be numerically approximated. Further note that this expression contains two hyperparameters, σ_ε and γ. σ_ε has been described above. γ is the scale parameter of the Cauchy distribution, which controls the heaviness of the tails of the distribution. In this case, larger values of γ increase the expected size of jumps in the utility function, corresponding to rapid and strong preference change, while smaller values correspond to a less rapidly changing utility function.

§6 Algorithmically Approximating The Utility Function

The update procedure can be approximated above in using a grid approximation for the PDF of the utility function, as described in the following algorithm, which I call the **adaptive** estimator for the utiltiy function.

1. Create an evenly spaced grid that ranges over a finite subset of the support of the PDF of $U(\cdot)$. Instead of computing the PDF over continuous real support, we will approximate the PDF of $U(\cdot)$ with a discrete-valued PMF with support over this grid.
2. Initialize the values of the grid using the approximated PDF of $U(0)$ according to some prior.
3. At each time step t:
4. Compute the prior PDF of $U(t)$ according to the step function Δ. This can be done by convolving the approximated PDF of $U(t-1)$ with f_Δ.
5. Observe either $X \succ_t \Upsilon_n$ or $X \prec_t \Upsilon_n$.
6. Depending on the observation, update the approximated PDF of $U(t)$ using $\pi(u,n)$ or $\pi(n,u)$ as the likelihood of the observation.

Furthermore, we can imagine two possible situations in which we hope to reconstruct utility functions. The first involves no intervention at all, where we are simply given a sequence of preferences $\left(X \underset{t}{\overset{?}{\succ}} \Upsilon_{n(t)}\right)$ for a given sequence of utilities $n(t)$. In the latter, we are able to pick $n(t)$ at each time and elicit a preference from the agent between X and $\Upsilon_{n(t)}$. In this case, choosing $n(t)$ to be the median of the approximated PDF of $U(t)$ will yield the highest information.

In the following section, I describe some computational results produced from an implementation of the above algorithm in Python. In my implementation, an `Agent` class is used to store $U(t)$ and iterate it stochastically according to the parameters σ_ε and γ provided to it. The `Agent` class also contains a `bid_result` method that returns the result of the `Agent`'s choice between X and $\Upsilon_{n(t)}$ based on the `Agent`'s current utility for X

(namely, $U(t)$) as well as the policy function $\pi(u, n(t))$ for the given $n(t)$. An iterative process is run for a specified number of steps. At each step, the median of the current grid-approximated prior PDF of $U(t)$ is used to make a bid to the `Agent`'s `bid_result` method, which is then used to observe the `Agent`'s preference between X and $\Upsilon_{n(t)}$. The new posterior for $U(t)$ is then computed using the `update_posterior` method. Then, the posterior for $U(t)$ is iterated using the step function f_Δ to yield the prior for $U(t+1)$, and the `Agent`'s actual utility is updated as well. At each time, the difference between the `Agent`'s true utility $U(t)$ and the median of the current posterior (i.e., the L_1 norm) is computed, and the sum of these errors is used as a measure of the cumulative accuracy of the PDF.

In my implementation, I made the additional simplifying assumption that utilities are strictly bounded on a given interval, centered at 0, to ensure that the grid approximation ranges sufficiently over the support of the PDF for the utility function. The interval can be represented as $[-L, L]$; I chose $L = 100$. This assumption is particularly problematic in the context of a Lévy flight, since it limits the degree to which large unpredictable jumps can occur, which is itself part of the motivation for this project. This limitation could be removed by using a particle filter instead of a grid, and this offers an opportunity for future work.

In addition, the approximation of the PDF of $U(t)$ uses values for σ_ε and γ that are not known in advance. In an application involving human agents, empirical methods might be used to estimate these values in advance, or a Maximum Likelihood Estimation procedure might be used to estimate them from observed preferences by attempting to predict the agent's preferences before they are revealed and selecting the values of σ_ε and γ that maximize correct predictions. However, it is important to note that the vNM theorem only states that utility functions are unique up to a linear transformation, and so the values of σ_ε and γ that maximize correct predictions are not necessarily the "true" values of σ_ε and γ: they are relative to the scaling of the utility function.

In my implementation, I did not use human preferences and instead used the `Agent` to generate results using given σ_ε and γ. To account for the fact that it is implausible to know the optimal values of σ_ε and γ in advance, I deliberately introduced random deviation between the values of σ_ε and γ that the `Agent` used and the ones used by the model. To do so, I multiplied the model's hyperparameters by a log-uniform parameter over the interval $[-1, 1]$:

$$\begin{aligned}
\sigma_\varepsilon^{\text{Agent}} &= \sigma_\varepsilon^{\text{Model}} * 10^{d_{\sigma_\varepsilon}} \\
\gamma^{\text{Agent}} &= \gamma^{\text{Model}} * 10^{d_\gamma} \\
d_{\sigma_\varepsilon}, d_\gamma &\sim \text{Uniform}(-1, 1)
\end{aligned}$$

Additionally, although Section 4 and Section 5 provide a statistcal methodology for updating $U(t)$ given $U(t-1)$ and an observation, it does not provide a prior for $U(0)$. In my implementation, I used a normal distribution with mean 0 and standard deviation $\frac{L}{2}$ (where the grid ranges over $[-L, L]$) as a prior for $U(0)$; this places 95% of the prior mass within the grid. A uniform distribution could have been used instead, and empirical data could have been used in real applications.

Finally, to comparatively evaluate the model's performance, I also implemented an estimator for the PDF of $U(t)$ that, broadly speaking, assumes perfect rationality and no preference change. The **rational** estimator assumes that the observation $X \succ_t \Upsilon_{n(t)}$ means that $U_{\text{timeless}} > n(t)$ and vice versa. Thus, the rational estimator simply truncates the PDF for U at the value of $n(t)$ and renormalizes. However, this leads to the possibility of inconsistent predictions, since observations $X \succ_{t_1} \Upsilon_a$ and $X \prec_{t_1} \Upsilon_b$ for $a > b$ together imply a contradiction: $U_{\text{timeless}} > a > b > U_{\text{timeless}}$. In terms of the estimated PDF, the posterior after updating on inconsistent preferences would have empty support and become an invalid probability function. An algorithm that assumed perfect rationality according to the vNM theorem would be unable to make any subsequent predictions, since a violation of its assumed axioms would have occurred. In order to continue making predictions, the rational estimator 'resets' its estimated PDF to a normal distribution centered at the current median with standard deviation $\frac{L}{2}$. The performance of the estimator specified in Section 5 can thus be compared to the performance of the rational estimator using the cumulative error.

Finally, as a visual aid for evaluating the estimators, I implemented an animated visualization containing three graphs:

- the estimated PDF of $U(t)$ at the current time step according to each estimator, labeled with the median of each estimated PDF and the true utility, with utility on the horizontal axis and probability density on the vertical axis
- the median of each estimator utility and the true utility over time, with time on the horizontal axis and utility on the vertical axis
- the cumulative error of each estimator over time, with time on the horizontal axis and cumulative error on the vertical axis

Having described the implementation of the model and the evalution procedure, I will proceed to describe some computational results.

§7 Computational Results

This section describes some computational results that compare the adaptive and rational estimators for the utility function using the above Python implementation. After I discuss the results of a single simulation, I proceed examine the results of 405 simulations over a range of values for σ_ε and γ. These results show that the adaptive estimator is significantly more effective at modeling the agent's utility function than the rational estimator. Keeping in mind that the so-called rational estimator is already significant more robust than the

§7.1 Example Iteration

Figure 1 is a screenshot of the three animated graphs described at the end of Section 6. In this iteration, the model uses $\sigma_\varepsilon = 1$ and $\gamma = 1$ (labeled as 'true values' on the first graph), while the agent's values have been perturbed: the agent uses $\sigma_\varepsilon = 3.211$ and $\gamma = 0.9841$.

1. The first graph shows that, at the current timestep, the adaptive posterior has much more of its mass concentrated around the true utility than the rational posterior (though this is not the case at all times in all iterations), and the adaptive posterior's median is much closer to the true utility than the rational posterior's median.
2. The second graph shows that although the rational estimator is roughly able to track the true utility, it often deviates with sharp spikes, typically caused by inconsistent preferences caused by preference change, error, or both. Note, in particular, that when the true utility changes rapidly, both the adaptive and rational estimators take some time to converge to the new true utility.
3. The third graph shows that the cumulative error of the adaptive posterior converges to a lower value than the cumulative error of the rational posterior over time. When the true utility does not change rapidly, the adaptive estimator performs significantly better than the rational estimator. When the true utility does change rapidly, both estimators incur signficant error while catching up to the new true utility.

Visual examination of this iteration suggests that the adaptive estimator is more effective at modeling the agent's utility function than the rational estimator. However, a more generalized empirical conclusion can only be drawn from a larger sample of simulations across different values of σ_ε and γ.

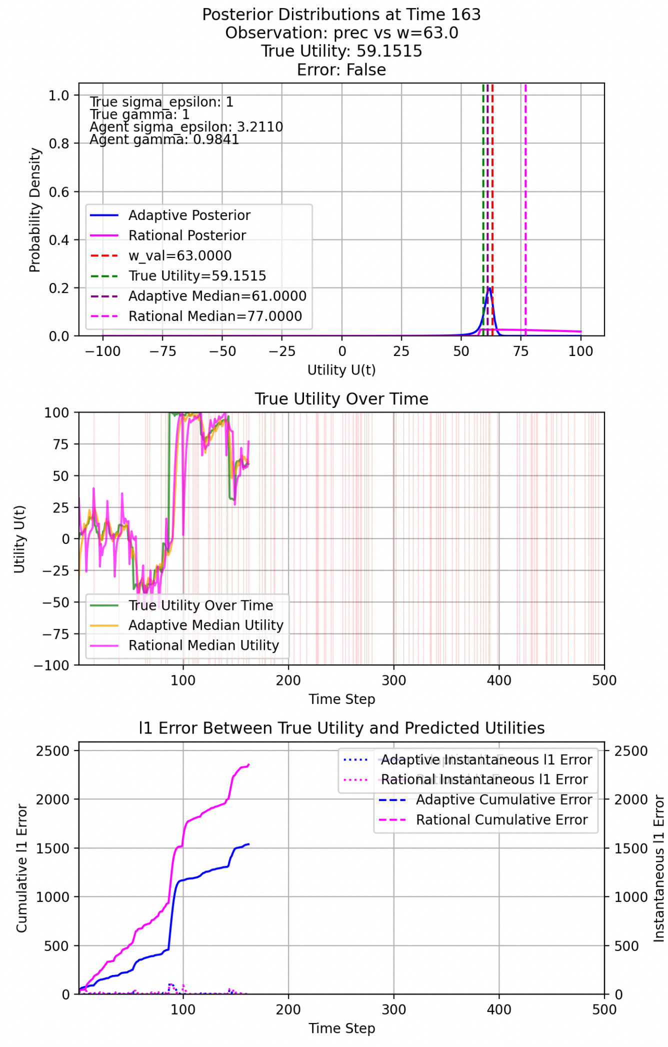

Figure 1. One simulation comparing the results of the two algorithms for estimating the utility function: the 'adaptive' algorithm which uses the sigmoid error policy and the Cauchy step size function, and the 'rational' algorithm which assumes perfect rationality and no preference change and which must therefore periodically reset its PDF for the utility.

§7.2 Brief Data Analysis

To evaluate the adaptive and rational estimators more broadly, I ran 405 simulations over a range of values for σ_ε and γ. In these simulations, the adaptive estimator used values for σ_ε and γ at each of 9 evenly log-spaced values from 0.1 to 10 (i.e., $10^{\{\frac{k}{4}-1\}}$ for integer

k ranging from 0 to 8), and five iterations were run at each pair of values. Each iteration included 500 time steps, and each iteration used new values for the agent's σ_ε and γ.

In these 405 iterations, the adaptive estimator's cumulative L_1 error was on average 2.38 times lower than the rational estimator's cumulative L_1 error. Larger values of γ increased the loss of each estimator, though the adaptive estimator's loss was still significantly lower than the rational estimator's loss (Figure 2). In addition, the adaptive estimator tended to outperform the rational estimator to a greater degree when the agent's value of γ was lower, i.e., when the agent's utility function changed less drastically (Figure 3).

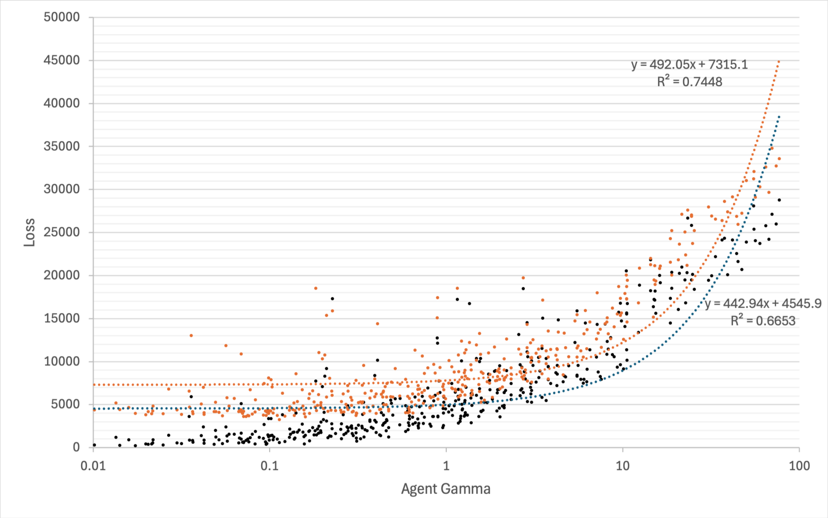

FIGURE 2. Each point represents the results of one estimator in one simulation. Orange points correspond to the rational estimator, and black points correspond to the adaptive estimator. The x-axis (log scale) represents the value of γ used by the agent, and the y-axis represents the cumulative error of the estimator. Trendlines are linear regressions of the points.

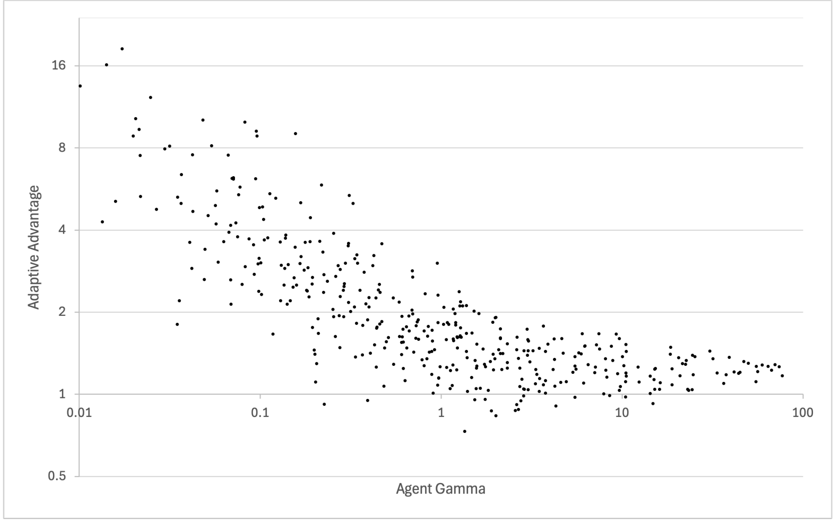

Figure 3. Each point represents the ratio between the cumulative error of the adaptive estimator and the cumulative error of the rational estimator in one simulation (rational divided by adaptive). The x-axis (log scale) represents the value of γ used by the agent, and the y-axis represents the ratio: higher values show a greater advantage for the adaptive estimator.

These results indicate that the adaptive estimator is significantly more effective at estimating the simulated agent's utility function than the rational estimator. In a way, this should be no surprise, since the simulated agent uses the same algorithm as the simulated agent, albeit often with perturbed parameters. However, the success of this model in this toy example may vindicate the general approach of using a Bayesian model to estimate the utility function of an agent outlined in general terms in Section 4. The model provided by Bayesian decision theory can be combined with empirical psychological results that provide other plausible error policies and step size functions.

§8 Conclusion

In this paper, I have introduced an adaptive estimator for utility functions that accounts for predictably irrational agents through a Bayesian framework. Unlike the traditional von Neumann-Morgenstern utility theory, which assumes fully rational agents with consistent preferences, my model accommodates the errors and preference changes observed in real-world decision-making.

As I argue in Section 1 and Section 3, both the the method of constructing a utility function given by the vNM theorem, as well as to Ng and Russell's approach to estimating utility functions using Inverse Reinforcement Learning and their formulation of a

Markov Decision problem, operate under the implausible assumption of a time-invariant utility framework and infallible evidence. The adaptive estimator I introduce employs a Bayesian approach that continuously updates the utility function in response to observed agent behaviors. This dynamic updating allows the model to account for both small fluctuating changes and large jumps in utility, helping to model the exhibited preference of predictably irrational agents. By explicitly modeling error and preference transitions, the adaptive estimator provides a method for estimating of an agent's utility that is both more theoretically flexible and empirically successful. Empirical results across numerous simulations demonstrate that the adaptive estimator consistently achieves significantly lower cumulative L_1 error compared to the rational estimator derived from traditional MDP frameworks. Although real agents likely do not make decisions in ways that can be perfectly described by the specific functions outlined in Section 5, the framework of the adaptive estimator introduced in Section 4 can accommodate any error policy and step size function.

One key limitation to the model I introduce is that it relies on the agent's hyperparameters, namely σ_ε and γ. I assume in the empirical results above that we can estimate the agent's hyperparameters within 1 order of magnitude. While my adaptive estimator performs well across a range of values under this assumption, determining the optimal settings in real-world scenarios remains a challenge. A real application of this model would require introducing methods to jointly estimate the agent's hyperparameters, both for error and preference change.

To recontextualize the results of this paper within the opening problem of AI alignment, I argue that the method of estimating the utility functions of an agent are significantly more psychologically and philosophically plausible than either the deterministic approach of the vNM theorem, which makes implausible assumptions of ideal rationality, or the oversimplified approach involving a linearity assumption in Ng and Russell's formulation of Inverse Reinforcement Learning. While inverse reinforcement learning may work reasonably well in cases where error and preference change are minimal or can be minimized, the adaptive model I introduced is better suited to cases where these phenomena may occur.

In particular, the success of inverse reinforcement learning within Reinforcement Learning with Human Feedback (RLHF) should not be taken to vindicate the methodology of inverse reinforcement learning in all cases. We can assume that most competent language users will converge to similar preferences about, for example, which of two LLM-generated responses is more helpful, and that these preferences do not change significantly over time (excluding cases where information becomes outdated; we might limit this claim to truly axiological change). However, AI alignment is a problem that is far broader than LLMs. Insofar as AI alignment involves furthering any particular human's preference-satisfaction, and that human preferences are subject to error and change, a successful strategy for AI alignment must be able to accommodate these phenomena. As we have seen in this paper, the adaptive estimator I introduced may provide a statistical methodology that can successfully achieve this task.

References

Gustafsson, Johan E. *Money-Pump Arguments.* Elements in Decision Theory and Philosophy. Cambridge, England: Cambridge University Press (Virtual Publishing), 2022

Hadfield-Menell, Dylan, Anca Dragan, Pieter Abbeel, and Stuart Russell. "The Off-Switch Game," 2016

Kripke, Saul A. *Wittgenstein on Rules and Private Language.* London, England: Harvard University Press, 1984

Ng, Andrew Y., and Stuart J. Russell. "Algorithms for Inverse Reinforcement Learning." In *Proceedings of the Seventeenth International Conference on Machine Learning*, 663–70. ICML '00. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2000

Russell, Stuart. *Human Compatible: Artificial Intelligence and the Problem of Control.* New York, NY: Penguin, 2019