Models of human preference for learning reward functions

Anonymous authors
Paper under double-blind review

Abstract

The utility of reinforcement learning is limited by the alignment of reward functions with the interests of human stakeholders. One promising method for alignment is to learn the reward function from human-generated preferences between pairs of trajectory segments, a type of reinforcement learning from human feedback (RLHF). These human preferences are typically assumed to be informed solely by partial return, the sum of rewards along each segment. We find this assumption to be flawed and propose modeling human preferences instead as informed by each segment's regret, a measure of a segment's deviation from optimal decision-making. Given infinitely many preferences generated according to regret, we prove that we can identify a reward function equivalent to the reward function that generated those preferences, and we prove that the previous partial return model lacks this identifiability property in multiple contexts. We empirically show that our proposed regret preference model outperforms the partial return preference model with finite training data in otherwise the same setting. Additionally, we find that our proposed regret preference model better predicts real *human* preferences and also learns reward functions from these preferences that lead to policies that are better human-aligned. Overall, this work establishes that the choice of preference model is impactful, and our proposed regret preference model provides an improvement upon a core assumption of recent research. We have open sourced our experimental code, the human preferences dataset we gathered, and our training and preference elicitation interfaces for gathering a such a dataset.

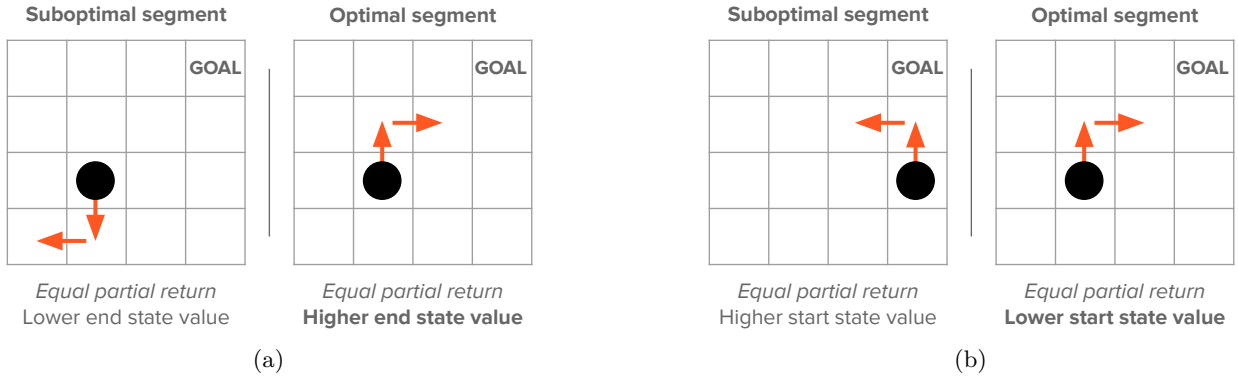

Figure 1: Illustrations of segment pairs for which the common partial return preference model poorly explains intuitive human preference. The task has -1 reward each time step, penalizing time taken to reach the goal. In both pairs, both segments have the same partial return (-2), but the one on the right is nonetheless the intuitive choice. However, the right segment in each pair consists only of optimal actions, whereas the left segment includes at least one suboptimal action. Regret, which our proposed preference model is based upon, is designed to measure a segment's deviation from optimal decision making. The right segment in each pair is therefore more likely to be preferred by a regret preference model. We suspect our human readers will also tend to prefer the right segments. (a) The preferred segment has a higher end state value. (b) The preferred segment has a lower start state value, indicating a lower opportunity cost (i.e., it did not waste a more valuable start state).

1 Introduction

Improvements in reinforcement learning (RL) have led to notable recent achievements (Silver et al., 2016; Senior et al., 2020; Vinyals et al., 2019; Bellemare et al., 2020; Berner et al., 2019; Degrave et al., 2022; Wurman et al., 2022), increasing its applicability to real-world problems. Yet, like all optimization algorithms, even *perfect* RL optimization is limited by the objective it optimizes. For RL, this objective is created in large part by the reward function. Poor alignment between reward functions and the interests of human stakeholders limits the utility of RL and may even pose risks of financial cost and human injury or death (Amodei et al., 2016; Knox et al., 2021).

Influential recent research has focused on learning reward functions from preferences over pairs of trajectory segments, a common form of reinforcement learning from human feedback (RLHF). Nearly all of this recent work assumes that human preferences arise probabilistically from *only* the sum of rewards over a segment, i.e., the segment's **partial return** (Christiano et al., 2017; Sadigh et al., 2017; Ibarz et al., 2018; Bıyık et al., 2021; Lee et al., 2021a;b; Ziegler et al., 2019; Wang et al., 2022; Ouyang et al., 2022; Bai et al., 2022; Glaese et al., 2022; OpenAI, 2022). That is, these works assume that people tend to prefer trajectory segments that yield greater accumulated rewards *during the segment*. However, this preference model ignores seemingly important information about the segment's desirability, including the state values of the segment's start and end states. Separately, this partial return preference model can prefer suboptimal actions with lucky outcomes, like buying a lottery ticket.

This paper proposes an alternative preference model based on the **regret** of each segment, which is a measure of how much each segment deviates from optimal decision-making. More precisely, regret is the negated sum of an optimal policy's advantage of each transition in the segment (Section 2.2). Figures 1 and 2 show intuitive examples of when these two models disagree. Some examples of domains where the preference models will differ are those with constant reward until the end, including competitive games like chess, go, and soccer as well as tasks for which the objective is to minimize time until reaching a goal.

For these two preference models, we first focus theoretically on a normative analysis (Section 3)—i.e., what preference model would we *want* humans to use if we could choose one based on how informative its generated preferences are—proving that reward learning on infinite, exhaustive preferences with our proposed regret preference model identifies a reward function with the same set of optimal policies as the reward function with which the preferences are generated. We also prove that the partial return preference model is not guaranteed to identify such a reward function in three different contexts: without preference noise, when trajectories of different lengths are possible from a state, and when segments consist of only one transition. We follow up with a descriptive analysis of how well each of these proposed models align with *actual* human preferences by collecting a human-labeled dataset of preferences in a rich grid world domain (Section 4) and showing that the regret preference model better predicts these human preferences (Section 5). Finally, we find that the policies ultimately created through the regret preference model tend to outperform those from the partial return model learning—both when assessed with collected human preferences or when assessed with synthetic preferences (Section 6). Our code for learning and for re-running our main experiments is publically available, alongside the human preferences dataset we gathered and our interface for training subjects and for preference elicitation.[1]

In summary, our primary contributions are five-fold:

1. We propose a new model for human preferences that is based on regret instead of partial return.
2. We theoretically validate that this regret-based model has the desirable characteristic of reward identifiability, and that the partial return model does not.
3. We empirically validate that when each preference model learns from a preferences dataset it created, this regret-based model leads to better-aligned policies.
4. We empirically validate that, with a collected dataset of human preferences, this regret-based model both better describes the human preferences and leads to better-aligned policies.
5. Overall, we show that the choice of preference model impacts the alignment of learned reward functions.

[1] References removed to maintain anonymity throughout the review process.

2 Preference models for learning reward functions

We assume that the task environment is a Markov decision process (MDP) specified by the tuple $(S, A, T, \gamma, D_0, r)$. S and A are the sets of possible states and actions, respectively. T is a transition function, $T: S \times A \rightarrow p(\cdot|s,a)$; γ is the discount factor; and D_0 is the distribution of start states. Unless otherwise stated, we assume all tasks are undiscounted (i.e., $\gamma = 1$) and have terminal states, after which only 0 reward can be received. Discounting is considered in depth in Appendix B.2. r is a reward function, $r: S \times A \times S \rightarrow \mathbb{R}$, where the reward r_t at time t is a function of s_t, a_t, and s_{t+1}. An MDP$\backslash r$ is an MDP without a reward function.

Throughout this paper, r refers to the ground-truth reward function for some MDP; $\hat{r}$ refers to a learned approximation of r; and $\tilde{r}$ refers to any reward function (including r or $\hat{r}$). A policy ($\pi: S \times A \rightarrow [0,1]$) specifies the probability of an action given a state. $Q^{\pi}_{\tilde{r}}$ and $V^{\pi}_{\tilde{r}}$ refer respectively to the state-action value function and state value function for a policy, π, under $\tilde{r}$, and are defined as follows.

$$V^{\pi}_{\tilde{r}}(s) \stackrel{\text{def}}{=} \mathbb{E}_{\pi}[\sum_{t=0}^{\infty} \tilde{r}(s_t, a_t, s_{t+1}) | s_0 = s]$$

$$Q^{\pi}_{\tilde{r}}(s,a) \stackrel{\text{def}}{=} \mathbb{E}_{\pi}[\tilde{r}(s,a,s') + V^{\pi}_{\tilde{r}}(s')]$$

An optimal policy π^* is any policy where $V^{\pi^*}_{\tilde{r}}(s) \geq V^{\pi}_{\tilde{r}}(s)$ at every state s for every policy π. We write shorthand for $Q^{\pi^*}_{\tilde{r}}$ and $V^{\pi^*}_{\tilde{r}}$ as $Q^*_{\tilde{r}}$ and $V^*_{\tilde{r}}$, respectively. The optimal advantage function is defined as $A^*_{\tilde{r}}(s,a) \triangleq Q^*_{\tilde{r}}(s,a) - V^*_{\tilde{r}}(s)$; this measures how much an action reduces expected return relative to following an optimal policy.

Throughout this paper, the ground-truth reward function r is used to algorithmically generate preferences when they are not human-generated, is hidden during reward learning, and is used to evaluate the performance of optimal policies under a learned $\hat{r}$.

2.1 Reward learning from pairwise preferences

A reward function can be learned by minimizing the cross-entropy loss—i.e., maximizing the likelihood—of observed human preferences, a common approach in recent literature (Christiano et al., 2017; Ibarz et al., 2018; Wang et al., 2022; Bıyık et al., 2021; Sadigh et al., 2017; Lee et al., 2021a;b; Ziegler et al., 2019; Ouyang et al., 2022; Bai et al., 2022; Glaese et al., 2022; OpenAI, 2022).

Segments Let σ denote a segment starting at state s^{σ}_0. Its length $|\sigma|$ is the number of transitions within the segment. A segment includes $|\sigma|+1$ states and $|\sigma|$ actions: $(s^{\sigma}_0, a^{\sigma}_0, s^{\sigma}_1, a^{\sigma}_1, ..., s^{\sigma}_{|\sigma|})$. In this problem setting, segments lack any reward information. As shorthand, we define $\sigma_t \triangleq (s^{\sigma}_t, a^{\sigma}_t, s^{\sigma}_{t+1})$. A segment σ is **optimal** with respect to $\tilde{r}$ if, for every $i \in \{1,...,|\sigma|\text{-}1\}$, $Q^*_{\tilde{r}}(s^{\sigma}_i, a^{\sigma}_i) = V^*_{\tilde{r}}(s^{\sigma}_i)$. A segment that is not optimal is **suboptimal**. Given some $\tilde{r}$ and a segment σ, $\tilde{r}^{\sigma}_t \triangleq \tilde{r}(s^{\sigma}_t, a^{\sigma}_t, s^{\sigma}_{t+1})$, and the undiscounted **partial return** of a segment σ is $\sum_{t=0}^{|\sigma|-1} \tilde{r}^{\sigma}_t$, denoted in shorthand as $\Sigma_{\sigma}\tilde{r}$.

Preference datasets Each preference over a pair of segments creates a sample $(\sigma_1, \sigma_2, \mu)$ in a preference dataset $D_{\succ}$. Vector $\mu = \langle \mu_1, \mu_2 \rangle$ represents the preference; specifically, if σ_1 is preferred over σ_2, denoted $\sigma_1 \succ \sigma_2$, $\mu = \langle 1,0 \rangle$. μ is $\langle 0,1 \rangle$ if $\sigma_1 \prec \sigma_2$ and is $\langle 0.5, 0.5 \rangle$ for $\sigma_1 \sim \sigma_2$ (no preference). For a sample $(\sigma_1, \sigma_2, \mu)$, we assume that the two segments have equal lengths (i.e., $|\sigma_1| = |\sigma_2|$).

Loss function To learn a reward function from a preference dataset, $D_{\succ}$, a common assumption is that these preferences were generated by a preference model P that arises from an unobservable *ground-truth* reward function r. We approximate r by minimizing cross-entropy loss to learn $\hat{r}$:

$$loss(\hat{r}, D_{\succ}) = -\sum_{(\sigma_1, \sigma_2, \mu) \in D_{\succ}} \mu_1 \log P(\sigma_1 \succ \sigma_2 | \hat{r}) + \mu_2 \log P(\sigma_1 \prec \sigma_2 | \hat{r}) \tag{1}$$

For a single sample where $\sigma_1 \succ \sigma_2$, the sample's likelihood is $P(\sigma_1 \succ \sigma_2 | \hat{r})$ and its loss is therefore $-log P(\sigma_1 \succ \sigma_2 | \hat{r})$. If $\sigma_1 \prec \sigma_2$, its likelihood is $1 - P(\sigma_1 \succ \sigma_2 | \hat{r})$. This loss is under-specified until $P(\sigma_1 \succ \sigma_2 | \hat{r})$ is defined, which is the focus of this paper. We show that the common partial return model of preference probabilities is flawed and introduce an improved regret-based preference model.

Preference models A preference model determines the probability of one trajectory segment being preferred over another, $P(\sigma_1 \succ \sigma_2|\tilde{r})$. $P(\sigma_1 \succ \sigma_2|\tilde{r})+P(\sigma_1 \sim \sigma_2|\tilde{r})+P(\sigma_1 \prec \sigma_2|\tilde{r})=1$, and $P(\sigma_1 \sim \sigma_2|\tilde{r})=0$ for the preference models considered herein. Preference models could be applied to model preferences provided by humans or other systems. Preference models can also directly generate preferences, and in such cases we refer to them as **preference generators**.

2.2 Choice of preference model: partial return and regret

Partial return All aforementioned recent work assumes human preferences are generated by a Boltzmann distribution over the two segments' partial returns, expressed here as a logistic function.[2]

$$P_{\Sigma r}(\sigma_1 \succ \sigma_2|\tilde{r}) = logistic\Big(\Sigma_{\sigma_1}\tilde{r} - \Sigma_{\sigma_2}\tilde{r}\Big). \tag{2}$$

Regret We introduce an alternative preference model based on the regret of each transition in a segment. We first focus on segments with deterministic transitions. For a transition (s_t,a_t,s_{t+1}) in a deterministic segment, $regret_{\mathrm{d}}(\sigma_t|\tilde{r}) \triangleq V^*_{\tilde{r}}(s^\sigma_t) - [\tilde{r}_t + V^*_{\tilde{r}}(s^\sigma_{t+1})]$. The subscript d in $regret_d$ signifies the assumption of deterministic transitions. For a full deterministic segment,

$$regret_d(\sigma|\tilde{r}) \triangleq \sum_{t=0}^{|\sigma|-1} regret_d(\sigma_t|\tilde{r}) = V^*_{\tilde{r}}(s^\sigma_0) - (\Sigma_\sigma \tilde{r} + V^*_{\tilde{r}}(s^\sigma_{|\sigma|})), \tag{3}$$

with the right-hand expression arising from cancelling out intermediate state values. Therefore, deterministic regret measures how much the segment reduces expected return from $V^*_{\tilde{r}}(s^\sigma_0)$. An optimal segment, σ^*, always has 0 regret, and a suboptimal segment, $\sigma^{\neg *}$, will always have positive regret, an intuitively appealing property that also plays a role in the identifiability proof of Theorem 3.1.

Stochastic state transitions, however, can result in $regret_d(\sigma^*|\hat{r}) > regret_d(\sigma^{\neg *}|\tilde{r})$, losing the property above. For instance, an optimal action can lead to worse return than a suboptimal action, based on stochasticity in state transitions. To retain this property that optimal segments have a regret of 0 and suboptimal segments have positive regret, we first note that the effect on expected return of transition stochasticity from a transition (s_t,a_t,s_{t+1}) is $[\tilde{r}_t + V^*_{\tilde{r}}(s_{t+1})] - Q^*_{\tilde{r}}(s_t,a_t)$ and add this expression once per transition to get $regret(\sigma)$, removing the subscript d that refers to determinism. The regret for a single transition becomes $regret(\sigma_t|\tilde{r}) = [V^*_{\tilde{r}}(s^\sigma_t) - [\tilde{r}_t + V^*_{\tilde{r}}(s^\sigma_{t+1})]] + [[\tilde{r}_t + V^*_{\tilde{r}}(s^\sigma_{t+1})] - Q^*_{\tilde{r}}(s^\sigma_t,a^\sigma_t)] = V^*_{\tilde{r}}(s^\sigma_t) - Q^*_{\tilde{r}}(s^\sigma_t,a^\sigma_t) = -A^*_{\tilde{r}}(s^\sigma_t,a^\sigma_t)$. Regret for a full segment is

$$regret(\sigma|\tilde{r}) = \sum_{t=0}^{|\sigma|-1} regret(\sigma_t|\tilde{r}) = \sum_{t=0}^{|\sigma|-1} \Big[V^*_{\tilde{r}}(s^\sigma_t) - Q^*_{\tilde{r}}(s^\sigma_t,a^\sigma_t)\Big] = \sum_{t=0}^{|\sigma|-1} -A^*_{\tilde{r}}(s^\sigma_t,a^\sigma_t). \tag{4}$$

The regret preference model is the Boltzmann distribution over negated regret:

$$P_{regret}(\sigma_1 \succ \sigma_2|\tilde{r}) \triangleq logistic\Big(regret(\sigma_2|\tilde{r}) - regret(\sigma_1|\tilde{r})\Big). \tag{5}$$

Lastly, we note that if two segments have deterministic transitions, end in terminal states, and have the same starting state, in this special case the regret model reduces to the partial return model: $P_{regret}(\cdot|\tilde{r}) = P_{\Sigma r}(\cdot|\tilde{r})$.

In this article, our *normative* results examine both tasks with deterministic transitions and tasks with stochastic transitions. These normative results include the theoretical analysis in Section 3 and the empirical results with synthetic data in Section 6.2 and Appendix F.2, with stochastic tasks specifically examined empirically in Appendix F.2.4. We gather human preferences for a deterministic task, which allows us to investigate the results with the more intuitive expression of $regret_d$ that includes partial return as a component.

[2]See Appendix B.1 for a derivation of this logistic expression from a Boltzmann distribution with a temperature of 1. Unless otherwise stated, we ignore the temperature because scaling reward has the same effect when preference probabilities are not deterministic. The temperature is allowed to vary for our theory in Section 3. Another context when the temperature parameter would be useful is when learning a single reward function with a loss function that includes one or more loss terms in addition to the formula in Equation 1; in such a case, scaling reward might undesirably affect the other loss term(s), whereas the varying the Boltzmann temperature changes the preference entropy without affecting the other loss term(s).

Algorithms in this paper All algorithms in the body of this paper can be summarized as "minimize Equation 1". They differ only in how the preference probabilities are calculated. All reward function learning via partial return uses Equation 2, replicating the dominant algorithm in recent literature (Christiano et al., 2017; Ibarz et al., 2018; Wang et al., 2022; Bıyık et al., 2021; Sadigh et al., 2017; Lee et al., 2021a;b; Ouyang et al., 2022). We use two algorithms for reward function learning via regret. The theory in Section 3 assumes exact measurement of regret, using Equation 5. Section 6 introduces Equation 6 to approximate regret—replacing Equation 5 to create another algorithm—and uses the resulting algorithm for our experimental results later in that section. Appendix B introduces other algorithms that use Equation 1, as well as one in Appendix B.4 that generalizes Equation 1.

Regret as a model for human preference P_{regret} makes at least three assumptions worth noting. First, it keeps the assumption that human preferences follow a Boltzmann distribution over some statistic, which is a common model of choice behavior in economics and psychology, where it is called the Luce-Shepard choice rule (Luce, 1959; Shepard, 1957). Second, P_{regret} implicitly assumes humans can identify optimal and suboptimal segments when they see them, which will be less true in domains where the human has less expertise. This assumption is similar to a common assumption of many algorithms for imitation learning, that humans can provide demonstrations that are optimal or noisily optimal (e.g., Abbeel & Ng (2004)).

Lastly, P_{regret} assumes that in stochastic settings where the best *outcome* may only result from suboptimal decisions (e.g., buying a lottery ticket), humans instead prefer optimal *decisions*. We suspect humans are capable of expressing either type of preference—based on decision quality or desirability of outcomes—and can be influenced by training or the preference elicitation interface. Curiously, for stochastic tasks in which preferences are based upon segments' observed outcomes, a preference model that uses deterministic $regret_d$ in Equation 5 appears fitting, since it does not subtract out the effects of fortunate and unfortunate transitions but does include segments' start and end state values.

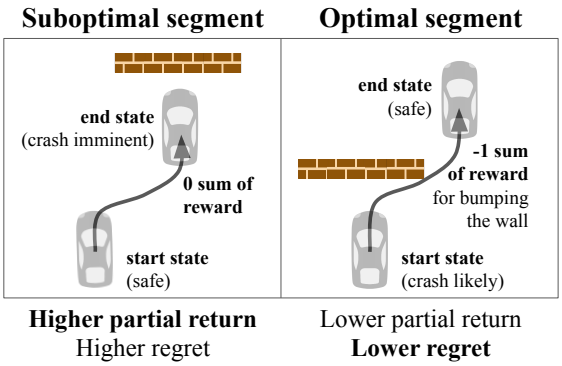

Figure 2: Two segments of a car moving at high speed near a brick wall. On the left, a car moves toward a brick wall; a bad crash is imminent, but has not yet occurred. On the right, a car escapes an imminent crash against a brick wall with only a scrape. Assume the right segment is optimal and the left segment is suboptimal (as defined in Sec. 2.1). The left segment has a higher sum of reward, so it is preferred under the partial return preference model. The right segment is preferred under the regret preference model since optimal segments have minimal regret. If we also assume deterministic transitions, then the regret model includes the difference in values between the start state and the end state (Equation 3), and the right segment would tend to be preferred because it greatly improves its state values from start to end, whereas the left segment's state values greatly worsen. We suspect our human readers will also tend to prefer the right segment.

In practice we determine that the regret model produces improvements over the partial return model (Section 6), and its assumptions represent an opportunity for follow-up research.

3 Theoretical comparison

In this section, we consider how different ways of *generating preferences* affect reward inference, setting aside whether humans can be influenced to give preferences in accordance with a specific preference method. In economics, this analysis—and all of our later analyses with synthetic preferences—could be considered a normative analysis. In artificial intelligence, this analysis might be cast as a step towards defining criteria for a rational preference model.

The theorems and proofs below focus on identifiability, a property which determines whether the parameters of a model can be recovered from infinite, exhaustive samples generated by the model. A model is unidentifiable if any two parameterizations of the model result in the same model behavior. In our setting, the model of concern is a preference model and the parameters constitute the ground-truth reward function, r. A preference model is identifiable if an infinite, exhaustive preferences dataset created by the preference model contains sufficient information to infer a behaviorally equivalent reward function to r. Note that *identifiability focuses on the preference model alone as a preference generator*, not on the learning algorithm that uses such a preference model.

This section uses preference models that include discounting (see Appendix B.2). We allow for discounting to make the theory more general and also because discounting is integral to Section 3.2.3. Here the notation for $Q^*_{\tilde{r}}(s,a)$ and $V^*_{\tilde{r}}(s)$ is expanded to $Q^*_{(\tilde{r},\tilde{\gamma})}(s,a)$ and $V^*_{(\tilde{r},\tilde{\gamma})}(s)$ respectively include the discount factor. To make the other content in this section specific to undiscounted tasks, simply assume all instances of $\tilde{\gamma}=1$, including the ground-truth γ and the $\hat{\gamma}$ used during reward function inference and policy improvement.

Definition 3.1 (An identifiable preference model)**.** *For a preference model P, assume an infinite dataset $D_{\succ}$ of pairs of segments is constructed by repeatedly choosing (σ_1,σ_2) and sampling a label $\mu \sim P(\sigma_1 \succ \sigma_2|r)$, using P as a preference generator. Further assume that, in this dataset, all possible segment pairs appear infinitely many times. For some M that is an MDP$\backslash(r,\gamma)$—an MDP with neither a reward function nor a discount factor—let $M_{(\tilde{r},\tilde{\gamma})}$ be M with the reward function $\tilde{r}$ and the discount factor $\tilde{\gamma}$. Let $\Pi^*_{(\tilde{r},\tilde{\gamma})}$ be the set of optimal policies for $M_{(\tilde{r},\tilde{\gamma})}$. Let problem-equivalence class $\mathfrak{R}$ be the set of all pairs of a reward function and a discount factor such that if $(r_1,\gamma_1),(r_2,\gamma_2) \in \mathfrak{R}$ then $\Pi^*_{(r_1,\gamma_1)} = \Pi^*_{(r_2,\gamma_2)}$. Preference model P is **identifiable** if and only if, for any choice of segment length n and ground-truth $M_{(r,\gamma)}$, there exists an operation on $D_{\succ}$ that always outputs a $(\hat{r},\hat{\gamma})$ that is in the same problem equivalence class as (r,γ). I.e., $\Pi^*_{(r,\gamma)} = \Pi^*_{(\hat{r},\hat{\gamma})}$.*

3.1 The regret preference model is identifiable.

We first prove that our proposed regret preference model is identifiable.

Theorem 3.1 (P_{regret} is identifiable)**.** *Let P_{regret} be any function such that if $regret(\sigma_1|\tilde{r},\tilde{\gamma}) < regret(\sigma_2|\tilde{r},\tilde{\gamma})$, $P_{regret}(\sigma_1 \succ \sigma_2|\tilde{r},\tilde{\gamma}) > 0.5$, and if $regret(\sigma_1|\tilde{r},\tilde{\gamma}) = regret(\sigma_2|\tilde{r},\tilde{\gamma})$, $P_{regret}(\sigma_1 \succ \sigma_2|\tilde{r},\tilde{\gamma}) = 0.5$. P_{regret} is identifiable.*

This class of regret preference models includes but is not limited to the Boltzmann distribution of Equation 5. Additionally, it includes a version of the regret preference model that noiselessly always prefers the segment with lower regret, as Theorem 3.2 considers for the partial return preference model.[3]

Consider reviewing the definitions of optimal segments and suboptimal segments in Section 2.1 before proceeding.

For the proof below, we will apply the following **sufficiency test for identifiability**. Preference model P is identifiable if, for any ground-truth $M_{(r,\gamma)}$, any $(\hat{r},\hat{\gamma}) = argmin_{(\tilde{r},\tilde{\gamma})}[loss(\tilde{r},\tilde{\gamma},D_{\succ})]$—for the cross-entropy loss (Eqn. 8, which is Eqn. 1 generalized to include discounting), with P as the preference model—is in the same problem equivalence class as (r,γ). I.e., $\Pi^*_{(r,\gamma)} = \Pi^*_{(\hat{r},\hat{\gamma})}$.

Proof Make all assumptions in Definition 3.1. Let $(\hat{r},\hat{\gamma}) = argmin_{(\tilde{r},\tilde{\gamma})}[loss(\tilde{r},\tilde{\gamma},D_{\succ})]$, where loss is the cross-entropy loss from Eqn. 8 with P_{regret} as the preference model.

Since $(\hat{r},\hat{\gamma})$ minimizes cross-entropy loss and is chosen from the complete space of reward functions and discount factors, $P_{regret}(\cdot|r,\gamma) = P_{regret}(\cdot|\hat{r},\hat{\gamma})$ for all possible segment pairs. Also, by Equation 12 (which generalizes Equation 4 to include discounting) $regret(\sigma|\tilde{r},\tilde{\gamma}) = 0$ if and only if σ is optimal with respect to $\tilde{r}$. And $regret(\sigma|\tilde{r},\tilde{\gamma}) > 0$ if and only if σ is suboptimal with respect to $(\tilde{r},\tilde{\gamma})$.

With respect to some $(\tilde{r},\tilde{\gamma})$, let σ^* be any optimal segment and $\sigma^{\neg *}$ be any suboptimal segment. $regret(\sigma^*|\tilde{r},\tilde{\gamma}) < regret(\sigma^{\neg *}|\tilde{r},\tilde{\gamma})$, so $P_{regret}(\sigma^* \succ \sigma^{\neg *}|\tilde{r},\tilde{\gamma}) > 0.5$. $P_{regret}(\cdot|\tilde{r},\tilde{\gamma})$ induces a total ordering over segments, defined by $regret(\sigma_1|\tilde{r},\tilde{\gamma}) < regret(\sigma_2|\tilde{r},\tilde{\gamma}) \iff P_{regret}(\sigma_1 \succ \sigma_2|\tilde{r},\tilde{\gamma}) > 0.5 \iff \sigma_1 > \sigma_2$ and $regret(\sigma_1|\tilde{r},\tilde{\gamma}) = regret(\sigma_2|\tilde{r},\tilde{\gamma}) \iff P_{regret}(\sigma_1 \succ \sigma_2|\tilde{r},\tilde{\gamma}) = 0.5 \iff \sigma_1 = \sigma_2$. Because regret has a minimum (0), there must be a set of segments which are ranked highest under this ordering, denoted $\Sigma^*_{(\tilde{r},\tilde{\gamma})}$. These segments in $\Sigma^*_{(\tilde{r},\tilde{\gamma})}$ are exactly those that achieve the minimum regret (0) and so are optimal with respect to $(\tilde{r},\tilde{\gamma})$.

Since the dataset $(D_{\succ})$ contains all segments by assumption, $\Sigma^*_{(\tilde{r},\tilde{\gamma})}$ contains all optimal segments with respect to $(\tilde{r},\tilde{\gamma})$. If a state-action pair (s,a) is in an optimal segment, then by the definition of an optimal segment $Q^*_{(\tilde{r},\tilde{\gamma})}(s,a) = V^*_{(\tilde{r},\tilde{\gamma})}(s)$. The set of optimal policies $\Pi^*_{\tilde{r}}$ for $\tilde{r}$ is all π such that, for all (s,a), if $\pi(s,a) > 0$, then $Q^*_{(\tilde{r},\tilde{\gamma})}(s,a) = V^*_{(\tilde{r},\tilde{\gamma})}(s)$. In short, $\Sigma^*_{(\tilde{r},\tilde{\gamma})}$ determines the set of each state-action pair (s,a) such that

[3] Equations 2 and 5 can be extended to include such noiseless preference models by including the temperature parameter of the Boltzmann distributions (after converting from their logistic formulations, reversing the derivation in Appendix B.1), where we assume that setting the temperature to 0 results in a hard maximum. In other words, when the temperature is 0 the preference is given deterministically to the segment with the higher partial return in Equation 2 or regret in Equation 5.

$Q^*_{(\tilde{r},\tilde{\gamma})}(s,a) = V^*_{(\tilde{r},\tilde{\gamma})}(s)$. This set determines $\Pi^*_{(\tilde{r},\tilde{\gamma})}$. Therefore $\Sigma^*_{(\tilde{r},\tilde{\gamma})}$ determines $\Pi^*_{(\tilde{r},\tilde{\gamma})}$, and we will refer to this determination as the function g.

We now focus on the reward function and discount factor used to generate preferences, (r,γ), and on the inferred reward function and discount factor, $(\tilde{r},\tilde{\gamma})$. Since $P_{regret}(\cdot|r,\gamma) = P_{regret}(\cdot|\hat{r},\hat{\gamma})$, (r,γ) and $(\hat{r},\hat{\gamma})$ induce the same total ordering over segments, and so $\Sigma^*_{(r,\gamma)} = \Sigma^*_{(\hat{r},\hat{\gamma})}$. Therefore $g(\Sigma^*_{(r,\gamma)}) = g(\Sigma^*_{(\hat{r},\hat{\gamma})})$. Since $g(\Sigma^*_{(r,\gamma)}) = \Pi^*_{(r,\gamma)}$ and $g(\Sigma^*_{(\hat{r},\hat{\gamma})}) = \Pi^*_{(\hat{r},\hat{\gamma})}$, $\Pi^*_{(r,\gamma)} = \Pi^*_{(\hat{r},\hat{\gamma})}$. □

The proof above establishes the identifiability of P_{regret} regardless of whether preferences are generated noiselessly or stochastically.

3.2 The partial return preference model is not generally identifiable.

In this subsection, we critique the previous standard preference model, the partial return model $P_{\Sigma r}$, by proving that this model can be unidentifiable in three different contexts.

- **Given *noiseless* preference labeling** by $P_{\Sigma r}$ in some MDPs, preferences never provide sufficient information to recover the set of optimal policies.
- **In variable-horizon tasks when the lengths of both segments in a pair are always equivalent.** Variable-horizon tasks include common tasks that terminate upon reaching success or failure states, reward functions that differ by a constant can have different sets of optimal policies. Yet for two such reward functions, the preference probabilities according to partial return will be identical.
- **With segment lengths of 1** ($|\sigma| = 1$), the discount factor γ does not affect the partial return preference model and therefore will not be recoverable from the preferences it generates. Since different values of γ can determine different sets of optimal policies, an inability to recover γ is a third type of unidentifiability.

We now prove in each of these three contexts that the partial return preference model is not identifiable.

For each, we will apply the following **sufficiency test for non-identifiability**. Preference model P is *not* identifiable if there exist two ground-truth MDPs, $M_{(r_1,\gamma_1)}$ and $M_{(r_2,\gamma_2)}$, such that $\Pi^*_{(r_1,\gamma_1)} \neq \Pi^*_{(r_2,\gamma_2)}$ and the infinite preference datasets created as described in Definition 3.1 by P for $M_{(r_1,\gamma_1)}$ and $M_{(r_2,\gamma_2)}$ are identical. Note that such identical preference datasets lack the information to differentiate which MDP they came from.

3.2.1 Partial return is not identifiable when preferences are noiseless.

Theorem 3.2 (Noiseless $P_{\Sigma r}$ is not identifiable)**.** *Let $P_{\Sigma r}$ be any function such that if $\Sigma_{\sigma_1}\tilde{r} > \Sigma_{\sigma_2}\tilde{r}$, $P_{\Sigma r}(\sigma_1 \succ \sigma_2|\tilde{r}) = 1$ and if $\Sigma_{\sigma_1}\tilde{r} = \Sigma_{\sigma_2}\tilde{r}$, $P_{\Sigma r}(\sigma_1 \succ \sigma_2|\tilde{r}) = 0.5$. There exists an MDP in which $P_{\Sigma r}$ is not identifiable.*

Below we present two proofs of Theorem 3.2. Each are proofs by counterexample. The first is a general proof and the second proof assumes a common characteristic, that all segments in the preference dataset are the same length. Though only one proof is needed, we present two because each counterexample demonstrates a qualitatively different category of how the partial return preference model can fail to identify the set of optimal policies.

Proof based on stochastic transitions: Assume the following class of MDPs, illustrated in Figure 3. The agent always begins at start state s_0. From s_0, action a_{safe} always transitions to s_{safe}, getting a reward of 0. From s_0, action a_{risk} transitions to s_{win} with probability 0.5, getting a reward of r_{win}, and transitions to s_{lose} with with probability 0.5, getting a reward of -10. In all MDPs in this class, $r_{win} > 0$. All 3 possible resulting states (s_{safe}, s_{win}, and s_{lose}) are absorbing states, from which all further reward is 0.

If $r_{win} \geq 10$, a_{risk} is optimal in s_0. If $r_{win} \leq 10$, a_{safe} is optimal in s_0. Three single-transition segments exist: (s_0,a_{safe},s_{safe}), (s_0,a_{risk},s_{win}), and (s_0,a_{risk},s_{lose}). By noiseless $P_{\Sigma r}$, $(s_0,a_{risk},s_{win}) \succ (s_0,a_{safe},s_{safe}) \succ (s_0,a_{risk},s_{lose})$, regardless of the value of r_{win}. In other words, $P_{\Sigma r}$ is insensitive to what the optimal action is from s_0 in this class of MDPs.

Now assume MDP M, where $r_{win} = 11$. In linear form, the weight vector for the reward function r_M can be expressed as $w_{r_{M_1}} = <-10,0,11>$. Let $\hat{r}_M$ have $w_{\hat{r}_M} = <-10,0,9>$. Both r_M and $\hat{r}_M$ have the same preferences as above, meaning that $\hat{r}_M$ minimizes loss on an infinite preferences dataset $D_{\succ}$ created by $P_{\Sigma r}$, yet it has a different optimal policy. Therefore, noiseless $P_{\Sigma r}$ is not identifiable. □

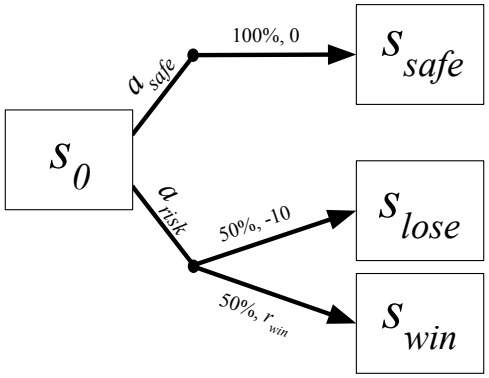

Figure 3: A class of MDPs in which, if $r_{win} > 0$, the partial return preference model fails the test for identifiability.

In contrast, note that by noiseless P_{regret}, the preferences are different than those above for $P_{\Sigma r}$. If $r_{win} > 10$, then $(s_0, a_{risk}, s_{win}) \sim (s_0, a_{risk}, s_{lose}) \succ (s_0, a_{safe}, s_{safe})$, If $r_{win} < 10$, then $(s_0, a_{safe}, s_{safe}) \succ (s_0, a_{risk}, s_{win}) \sim (s_0, a_{risk}, s_{lose})$. Intuitively, this difference comes from P_{regret} always giving higher preference probability to optimal actions, even if they result in bad outcomes. Another perspective can be found from the utility theory of Von Neumann & Morgenstern (1944). Specifically, $P_{\Sigma r}$ gives preferences over outcomes, which in the terms of utility theory can only be used to infer an ordinal utility function. Ordinal utility functions are merely consistent with the preference ordering over outcomes and do not generally capture preferences over actions when their outcomes are stochastically determined. The deterministic regret preference model, P_{regret_d}, also has this weakness in tasks with stochastic transitions. On the other hand, P_{regret} forms preferences over so-called lotteries—the distribution over possible outcomes—and can therefore learn a cardinal utility function, which can explain preferences over risky action. See Russell & Norvig (2020, Ch. 16) for more detail on these concepts from expected utility theory.

Since the proof above focuses upon stochastic transitions, we show the lack of identifiability for noiseless $P_{\Sigma r}$ can be found for quite different reasons in a deterministic MDP when the preferences dataset has a common characteristic: that all segments are the same length.

Proof based on segments of fixed length: Consider the MDP M_1 in Figure 4 and assume preferences are given over segments with length 1 (i.e., containing one transition). The optimal policy for M_1 is to move rightward from s_0, whereas optimal behavior for M_1' is to move downward from s_0. In *both* M_1 and M_1', preferences by $P_{\Sigma r}$ are as follows, omitting the action for brevity: $(s_a, s_0) \sim (s_a, s_{term}) \sim (s_0, s_a) \succ (s_0, s_{term})$. As in the previous proof, $P_{\Sigma r}$ is insensitive to certain changes in the reward function that alter the set of optimal policies. Whenever this characteristic is found, $\Pi_r^* = \Pi_{\hat{r}}^*$ is not guaranteed, failing the test for identifiability. Here specifically, the reward function for M_1' would achieve the minimum possible cross-entropy loss on an exhaustive preference dataset created in M_1 with the noiseless preferences from the partial return preference model, despite the optimal policy in M_1' conflicting with the ground-truth optimal policy.

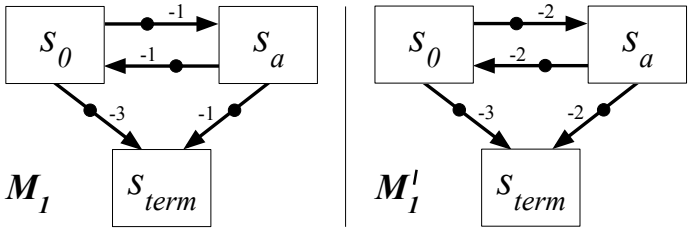

Figure 4: An MDP (M_1) where $\Pi_r^* = \Pi_{\hat{r}}^*$ is not guaranteed for the partial return preference model, failing the test for identifiability with segments of length 1. The ground-truth reward function is shown to the left, and an MDP M_1' with an alternative reward function is shown to the right. Under partial return, both create the same set of preferences despite having different optimal actions from s_0.

The logic of this proof can be applied for trajectories of length 2 in the MDP M_2 shown in Figure 5. Together, M_1 and M_2 suggest a rule for constructing an MDP where $\Pi_r^* = \Pi_{\hat{r}}^*$ is not guaranteed for $P_{\Sigma r}$, failing the identifiability test for any fixed segment length, $|\sigma|$: set the number of states to the right of s_0 to $|\sigma|$ (not counting s_{term}), set the reward r_{fail} for (s_0, s_{term}) such that $r_{fail} < 0$, and set the reward for each other transition to $c + r_{fail}/(|\sigma|+1)$, where $c > 0$. Given an MDP constructed this way, an alternative reward function that results in the same preferences under $P_{\Sigma r}$ yet has a different optimal action from s_0 can then be constructed by changing all reward other than r_{fail} to $c + r_{fail}/(|\sigma|+1)$, where c now is constrained to $c < 0$ and $c \times |\sigma| < r_{fail}$.

Note that the set of preferences for each of these MDPs is the same even when including segments that reach terminal state before $|\sigma|$ transitions (which can still be considered to be of length $|\sigma|$ if the terminal state is an absorbing state from which reward is 0).

Discussion of preference noise and identifiability Of the two proofs by example for Theorem 3.2, the first proof's example reveals issues when learning reward functions with stochastic transitions with either $P_{\Sigma r}$ or *deterministic* P_{regret_d}. These issues directly correspond to the need for preferences over distributions over outcomes (i.e., lotteries) to construct a cardinal utility function (see Russell & Norvig (2020, Ch. 16)). Correspondingly, when Skalse et al. (2022) consider reward identifiability with the partial return preference model, they change the learning problem such that a training sample consists of preferences between *distributions* over trajectories. Intuitively, Theorem 3.2 says that $P_{\Sigma r}$ is not identifiable without the distribution over preferences providing information about the proportions of rewards with respect to each other. In contrast, to be identifiable, the regret preference model does not require this preference error (though it can presumably benefit from it in certain contexts).

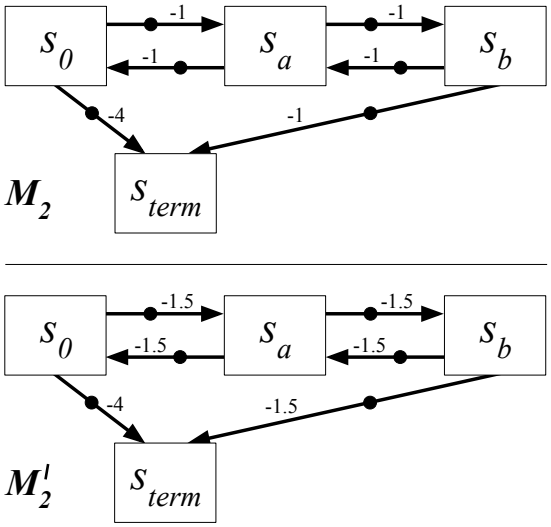

Figure 5: An MDP (M_2) where $\Pi_r^* = \Pi_{\hat{r}}^*$ is not guaranteed for the partial return preference model, failing the test for identifiability with segments of length 2. The ground-truth reward function is shown in the top diagram, and an MDP M_2' with an alternative reward function is shown in the bottom diagram. Under partial return, both create the same set of preferences despite having different optimal actions from s_0.

3.2.2 Partial return is not identifiable in variable-horizon tasks.

Many common tasks have the characteristic of having at least one state from which trajectories of *different* lengths are possible, which we refer to as being a **variable-horizon task**. Tasks that terminate upon completing a goal typically have this characteristic. We also assume that for any pair of segments in the preference dataset, the lengths of those two segments are equivalent. This assumption describes typical practice (e.g., Christiano et al. (2017); Sadigh et al. (2017); Ibarz et al. (2018); Bıyık et al. (2021); Lee et al. (2021a); Wang et al. (2022)). In this context, we show another way that the partial return preference model is not identifiable, a limitation that has arisen dramatically in our own experiments and is not limited to noiseless preferences: *adding a constant to a reward function will often change the set of optimal policies, but it will not change the probability of preference for any two segments. Therefore, those preferences will not contain information to recover the set of optimal policies.*

We now explain why such a constant shift will not change the probability of preference based upon partial return. Consider a constant value c and two reward functions, r_1 and r_2, where $r_1(s_t,a_t,s_{t+1}) - r_2(s_t,a_t,s_{t+1}) = c$ for all transitions (s_t,a_t,s_{t+1}). The partial return of any segment of length $|\sigma|$ will be $c \times |\sigma|$ higher for r_1 than for r_2 (assuming an undiscounted task, $\gamma = 1$). In the partial return preference model (Equation 2), this addition of $c \times |\sigma|$ to each segment's partial return cancels out, having no effect on the different in the segments' partial returns and therefore also having no effect on the preference probabilities. Consequently, adding c to a reward function's output will also not affect the distribution over preference datasets that the partial return preference model would create via sampling from its preference probabilities.

If, for each state in an MDP, all possible trajectories from that state have the *same* length, then adding a constant c to the output of the reward function does not affect the set of optimal policies. Specifically, the set of optimal policies is preserved because the return for any trajectory from a state is changed by $c \times |\tau|$, where $|\tau|$ is the unchanging trajectory length from that state, so the ranking of trajectories by their returns is unchanged and also the expected return of a policy from that state is unchanged. Continuing tasks and fixed-horizon tasks have this property.

However, if trajectories from a state can terminate after *different* numbers of transitions, then two reward functions that differ by a constant can have different sets of optimal policies. Episodic tasks are often vulnerable

to this issue. To illustrate, consider the task in Figure 1, a simple grid world task that penalizes the agent with -1 reward for each step it takes to reach the goal. If this reward per step is shifted to $+1$ (or any positive value), then any optimal policy will *avoid the goal*, flipping the objective of the task from that of reaching the goal. So, for variable-horizon tasks, $P_{\Sigma r}$ is not generally identifiable.

Though identifiability focuses on what information is encoded in preferences, this issue has practical consequences during *learning* from preferences over segments of length 1: for a preferences dataset, all reward functions that differ by a constant assign the same likelihood to the dataset, making the choice between such reward functions arbitrary and the learning problem underspecified. Some past authors have acknowledged this insensitivity to a shift (Christiano et al., 2017; Lee et al., 2021a; Ouyang et al., 2022; Hejna & Sadigh, 2023), and the common practice of forcing all tasks to have a fixed horizon (Christiano et al., 2017; Gleave et al., 2022) may be partially attributable to $P_{\Sigma r}$'s lack of identifiability in variable-horizon tasks, leading to its low performance in such tasks. In Appendix F.2.2, we propose a stopgap solution to this problem and also observe that in episodic grid worlds that the partial return preference model performs catastrophically poorly without this solution, both with synthetic preferences and human preferences.

The regret preference model is appropriately affected by constant reward shifts. Here we give intuition for why adding a constant c to the output of a reward function does not threaten the identifiability of the regret preference model, as established in Theorem 3.1. As stated above, adding c to reward can change the set of optimal policies. Any such change in what actions are optimal would likewise change the ordering of segments by regret, so the likelihood of a preferences dataset according to the regret preference model *would* be affected by such a constant shift in the learned reward function (as it should be).

3.2.3 Partial return is not identifiable for segment lengths of 1.

Arguably the most impactful application to date of learning reward functions from human preferences is to fine-tune large language models. For the most notable of these applications, the segment length $|\sigma|=1$ (Ziegler et al., 2019; OpenAI, 2022; Glaese et al., 2022; Ouyang et al., 2022; Bai et al., 2022).

Changing γ often changes the set of optimal policies, yet when $|\sigma|=1$, changing the discount factor does not change preference probabilities based upon partial return preference model. We elaborate below.

Here we make an exception to this article's default assumption that all tasks are undiscounted. As we describe in Appendix B.2, the discounted partial return of a segment is $\sum_{t=0}^{|\sigma|-1}\tilde{\gamma}^t\tilde{r}_t^\sigma$. We follow the standard convention that $0^0=1$. When $|\sigma|=1$, the partial return simplifies to the immediate reward, $\tilde{r}_0^\sigma$, regardless of the value of γ. Consequently the partial return preference model is unaffected by the discount factor when $|\sigma|=1$. We leave to the reader the task of a precise proof by counterexample that partial return is not identifiable when $|\sigma|=1$; any two MDPs that differ only by their discount factor and have different sets of optimal policies will suffice to provide a proof, since the distribution of preferences according to partial return will be identical in each of these MDPs, establishing the lack of identifiability.

To remove this source of unidentifiability, the preferences dataset would need to be presented to the learning algorithm with a corresponding discount factor. Past work on identifiability in this setting (Skalse et al., 2022) has assumed that the discount factor is given and does not discuss the topic further.

As with the other identifiability issues demonstrated in this subsection, this issue has practical consequences *during learning* from preferences. When $|\sigma|=1$, the choice of $\hat{\gamma}$ is arbitrary, making the learning problem underspecified.

The regret preference model is identifiable even when the discount factor is unknown. Note that Theorem 3.1 already includes this case. To add some intuition, the discounted regret of a segment—presented in Appendix B.2—does include the discount factor in its formulation, regardless of segment length. Therefore, the discount factor used during preference generation does impact what reward function is learned.

4 Creating a human-labeled preference dataset

To empirically investigate the consequences of each preference model when learning reward from *human* preferences, we collected a preference dataset labeled by human subjects via Amazon Mechanical Turk. This data collection was IRB-approved. Appendix D adds detail to the content below.

4.1 The general delivery domain

The delivery domain consists of a grid of cells, each of a specific road surface type. The delivery agent's state is its location. The agent's action space is moving one cell in one of the four cardinal directions. The episode can terminate either at the destination for $+50$ reward or in failure at a sheep for -50 reward. The reward for a non-terminal transition is the sum of any reward components. Cells with a white road surface have a -1 reward component, and cells with brick surface have a -2 component. Additionally, each cell may contain a coin ($+1$) or a roadblock (-1). Coins do not disappear and at best cancel out the road surface cost. Actions that would move the agent into a house or beyond the grid's perimeter result in no motion and receive reward that includes the current cell's surface reward component but not any coin or roadblock components. In this work, the start state distribution, D_0, is always uniformly random over non-terminal states. This domain was designed to permit subjects to easily identify bad behavior yet also to be difficult for them to determine *optimal* behavior from most states, which is representative of many common tasks. Note that this intended difficulty forces some violation of the regret preference model's assumption that humans always prefer optimal segments over suboptimal ones, therefore testing its performance in non-ideal conditions.

4.1.1 The delivery task

We chose one instantiation of the delivery domain for gathering our dataset of human preferences. This specific MDP has a 10×10 grid. From every state, the highest return possible involves reaching the goal, rather than hitting a sheep or perpetually avoiding termination. Figure 6 shows this task.

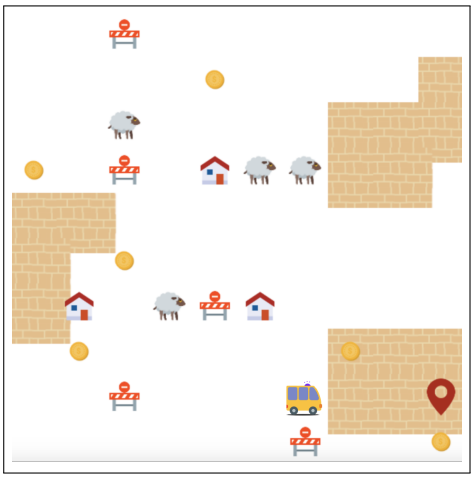

Figure 6: The delivery task used to gather human preferences. The yellow van is the agent and the red inverted teardrop is the destination.

4.2 The subject interface and survey

This subsection describes the three main stages of each data collection session. A video showing the full experimental protocol can be seen at bit.ly/humanprefs.

Teaching subjects about the task Subjects first view instructions describing the general domain. To avoid the jargon of "return" and "reward," these terms are mapped to equivalent values in US dollars, and the instructions describe the goal of the task as maximizing the delivery vehicle's financial outcome, where the reward components are specific financial impacts. This information is shared amongst interspersed interactive episodes, in which the subject controls the agent in domain maps that are each designed to teach one or two concepts. Our intention during this stage is to inform the later preferences of the subject by teaching them about the domain's dynamics and its reward function, as well as to develop the subject's sense of how desirable various behaviors are. At the end of this stage, the subject controls the agent for two episodes in the specific delivery task shown in Figure 6.

Preference elicitation After each subject is trained to understand the task, they indicate their preferences between 40–50 randomly-ordered pairs of segments, using the interface shown in Figure 7. The subjects select a preference, no preference ("same"), or "can't tell". In this work, we exclude responses labeled "can't tell", though one might alternatively try to extract information from these responses.

Subjects' task comprehension Subjects then answered questions testing their understanding of the task, and we removed their data if they scored poorly. We also removed a subject's data if they preferred colliding

the vehicle into a sheep over not doing so, which we interpreted as poor task understanding or inattentiveness. This filtered dataset contains 1812 preferences from 50 subjects.

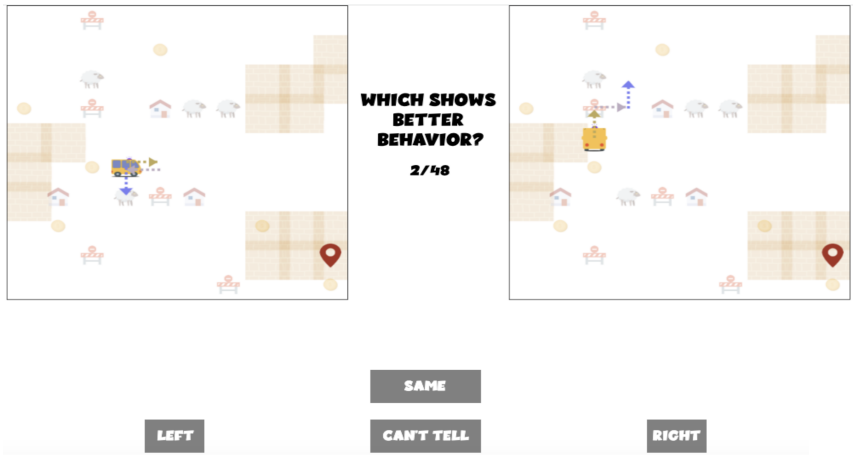

Figure 7: Interface shown to subjects during preference elicitation. The left trajectory shows the yellow van doubling back on itself before hitting a sheep. The right trajectory shows the van hitting a road block.

4.3 Selection of segment pairs for labeling

We collected human preferences in two stages, each with different methods for selecting which segment pairs to present for labeling. The sole purpose of collecting this second-stage data was to improve the reward-learning performance of the partial return model, $P_{\Sigma r}$. Without second-stage data, $P_{\Sigma r}$ compared even worse to P_{regret} than in the results described in Section 6, performing worse than a uniformly random policy (see Appendix F.3.3). Both stages' data are combined and used as a single dataset. These methods and their justification are described in Appendix D.3.

5 Descriptive results

This section considers how well different preference models explain our dataset of human preferences.

5.1 Correlations between preferences and segment statistics

Recall that with deterministic transitions, the regret of a segment has 3 components: $regret_d(\sigma|\tilde{r}) = V^*_{\tilde{r}}(s^\sigma_0) - (\Sigma_\sigma \tilde{r} + V^*_{\tilde{r}}(s^\sigma_{|\sigma|}))$, one of which is partial return, $\Sigma_\sigma \tilde{r}$. We hypothesize that the other two terms—the values of segments' start and end states, which are included in P_{regret} but not in P_Σ—affect human preferences, independent of partial return. If this hypothesis is true, then we have more confidence that preference models that include start and end state values will be more effective during inference of the reward functions.

The dataset of preferences is visualized in Figure 8. To simplify analysis, we combine the two parts of $regret_\mathrm{d}(\sigma|r)$ that are additional to $\Sigma_\sigma \tilde{r}$ and introduce the following shorthand: $\Delta_\sigma V_{\tilde{r}} \triangleq V^*_{\tilde{r}}(s^\sigma_{|\sigma|}) - V^*_{\tilde{r}}(s^\sigma_0)$.

Note that with an algebraic manipulation (see Appendix E.1), $regret_\mathrm{d}(\sigma_2|\tilde{r}) - regret_\mathrm{d}(\sigma_1|\tilde{r}) = (\Delta_{\sigma_1} V_{\tilde{r}} - \Delta_{\sigma_2} V_{\tilde{r}}) + (\Sigma_{\sigma_1} \tilde{r} - \Sigma_{\sigma_2} \tilde{r})$. Therefore, on the diagonal line in Figure 8, $regret_\mathrm{d}(\sigma_2|r) = regret_\mathrm{d}(\sigma_1|r)$, making the P_{regret_d} preference model indifferent. This plot shows how $\Delta_\sigma V_r$ has influence independent of partial return by focusing only on points at a chosen y-axis value; if the colors along the corresponding horizontal line reddens as the x-axis value increases, then $\Delta_\sigma V_r$ appears to have independent influence.

To statistically test for independent influence of $\Delta_\sigma V_r$ on preferences, we consider subsets of data where $\Sigma_{\sigma_1} r - \Sigma_{\sigma_2} r$ is constant. For $\Sigma_{\sigma_1} r - \Sigma_{\sigma_2} r = -1$ and $\Sigma_{\sigma_1} r - \Sigma_{\sigma_2} r = -2$, the only values with more than 30 samples that also include informative samples with both negative and positive values of $regret(\sigma_1|r) - regret(\sigma_2|r)$, the Spearman's rank correlations between $\Delta_\sigma V_r$ and the preferences are significant ($r >= 0.3$, $p < 0.0001$). This

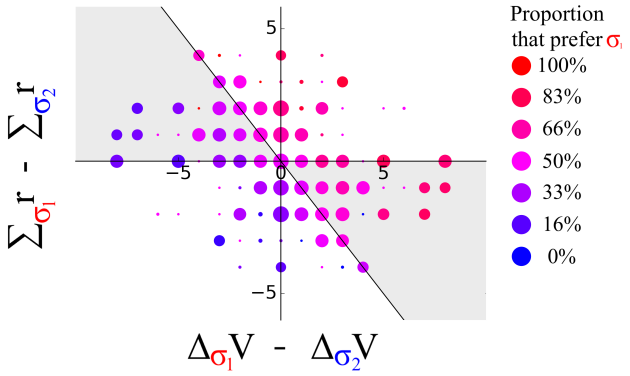

Figure 8: Proportions at which subjects preferred each segment in a pair, plotted by the difference in the segments' changes in state values (x-axis) and partial returns (y-axis). The diagonal line shows points of preference indifference for P_{regret}. Points of indifference for P_{Σ} lie on the x-axis. The shaded gray area indicates where the partial return and regret models disagree, each giving a different segment a preference probability greater than 0.5. Each circle's area is proportional to the number of samples it describes. A visual test of which preference model better fits the data is as follows: if the human subjects followed the partial return preference model, the color gradient would be orthogonal to the x-axis. If they followed the regret preference model, the color gradient would be orthogonal to the diagonal line, since regret on this plot is $x+y$. Visual inspection of the color gradient reveals the latter color gradient, suggesting they more closely followed the regret preference model.

result indicates that $\Delta_\sigma V_r$ *influences human preferences independent of partial return*, validating our hypothesis that humans form preferences based on information about segments' start states and end states, not only partial returns.

5.2 Likelihood of human preferences under different preference models

To examine how well each preference model predicts human preferences, we calculate the cross-entropy loss for each model (Equation 1)—i.e., the negative log likelihood—of the preferences in our dataset. Scaling reward by a constant factor does not affect the set of optimal policies. Therefore, throughout this work we ensure that our analyses of preference models are insensitive to reward scaling. To do so for this specific analysis, we conduct 10-fold cross validation to learn a reward scaling factor for each of P_{regret} and $P_{\Sigma r}$. Table 1 shows that the loss of P_{regret} is lower than that of $P_{\Sigma r}$, indicating that P_{regret} is more reflective of how people actually express preferences.

Preference model	Loss
$P(\cdot)=0.5$ (uninformed)	0.69
$P_{\Sigma r}$ (partial return)	0.62
P_{regret}	**0.57**

Table 1: Mean cross-entropy test loss over 10-fold cross validation (n=1812) from predicting human preferences. Lower is better.

6 Results from learning reward functions

Analysis of a preference model's predictions of human preferences is informative, but such predictions are a means to the ends of learning human-aligned reward functions and policies. We now examine each preference model's performance in these terms. In all cases, we learn a reward function $\hat{r}$ according to Equation 1 and apply value iteration (Sutton & Barto, 2018) to find the approximately optimal $Q^*_{\hat{r}}$ function. For this $Q^*_{\hat{r}}$, we then evaluate the mean return of the maximum-entropy optimal policy—which chooses uniformly randomly among all *optimal* actions—with respect to the ground-truth reward function r, over D_0. This methodology is illustrated in Figure 9.

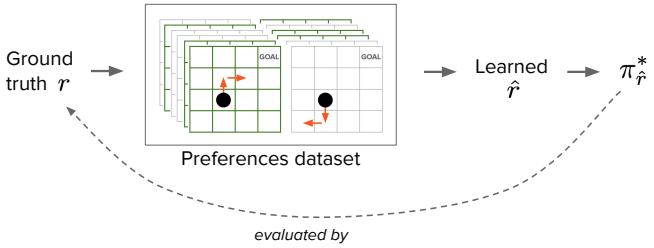

Figure 9: The general design pattern used for learning a reward function from preferences and evaluating that reward function. The generic gridworld shown is for illustrative purposes only.

To compare performance across different MDPs, the mean return of a policy π, V_r^π, is normalized to $(V_r^\pi - V_r^U)/V_r^*$, where V_r^* is the optimal expected return and V_r^U is the expected return of the uniformly random policy (both given D_0). Normalized mean return above 0 is better than V_r^U. Optimal policies have a normalized mean return of 1, and we consider above 0.9 to be *near optimal*.

6.1 An algorithm to learn reward functions with $regret(\sigma|\hat{r})$

Algorithm 1 is a general algorithm for learning a *linear* reward function according to P_{regret}. This regret-specific algorithm only changes the regret-based algorithm from Section 2.2 by replacing Equation 5 with a tractable approximation of regret, avoiding expensive repeated evaluation of $V_{\hat{r}}^*(\cdot)$ and $Q_{\hat{r}}^*(\cdot,\cdot)$ to compute $P_{regret}(\cdot|\hat{r})$ during reward learning. Specifically, successor features for a set of policies are used to approximate the optimal state values and state-action values for *any* reward function. This algorithm straightforwardly applies general policy iteration (GPI) with successor features to approximate optimal state and action values for arbitrary reward functions, as described by Barreto et al. (2016).

Approximating P_{regret} with successor features Following the notation of Barreto et al., assume the ground-truth reward is linear with respect to a feature vector extracted by $\boldsymbol{\phi}: S \times A \times S \rightarrow \mathbb{R}^d$ and a weight vector $\boldsymbol{w_r} \in \mathbb{R}^d$: $r(s,a,s') = \boldsymbol{\phi}(s,a,s')^\top \boldsymbol{w_r}$. During learning, $\boldsymbol{w_{\hat{r}}}$ similarly expresses $\hat{r}$ as $\hat{r}(s,a,s') = \boldsymbol{\phi}(s,a,s')^\top \boldsymbol{w_{\hat{r}}}$.

Given a policy π, the successor features for (s,a) are the expectation of discounted reward features from that state-action pair when following π: $\boldsymbol{\psi_Q^\pi}(s,a) = E^\pi[\sum_{t=0}^\infty \gamma^t \boldsymbol{\phi}(s_t,a_t,s_{t+1})|s_0 = s, a_0 = a]$. Therefore, $Q_{\hat{r}}^\pi(s,a) = \boldsymbol{\psi_Q^\pi}(s,a)^\top \boldsymbol{w}_{\hat{r}}$. Additionally, state-based successor features can be calculated from the $\boldsymbol{\psi_Q^\pi}$ above as $\boldsymbol{\psi_V^\pi}(s) = \sum_{a \in A} \pi(a|s) \boldsymbol{\psi_Q^\pi}(s,a)$, making $V_{\hat{r}}^\pi(s) = \boldsymbol{\psi_V^\pi}(s)^\top \boldsymbol{w}_{\hat{r}}$.

Given a set Ψ_Q of state-action successor feature functions and a set Ψ_V of state successor feature functions for various policies and given a reward function via $\boldsymbol{w}_{\hat{r}}$, $Q_{\hat{r}}^{\pi^*}(s,a) \geq max_{\boldsymbol{\psi_Q} \in \Psi_Q}[\boldsymbol{\psi_Q^\pi}(s,a)^\top \boldsymbol{w}_{\hat{r}}]$ and $V_{\hat{r}}^{\pi^*}(s) \geq max_{\boldsymbol{\psi_V} \in \Psi_V}[\boldsymbol{\psi_V^\pi}(s)^\top \boldsymbol{w}_{\hat{r}}]$ (Barreto et al., 2016), so we use these two maximizations as approximations of $Q_{\hat{r}}^*(s,a)$ and $V_{\hat{r}}^*(s)$, respectively. In practice, to enable gradient-based optimization with current tools, the maximization in this expression is replaced with the softmax-weighted average, making the loss function linear. Focusing first on the approximation of $V_{\hat{r}}^*(s)$, for each $\boldsymbol{\psi_V} \in \Psi_V$, a softmax weight is calculated for $\boldsymbol{\psi_V^\pi}(s)$: $softmax_{\Psi_V}(\boldsymbol{\psi_V^\pi}(s)^\top \boldsymbol{w}_{\hat{r}}) \triangleq [(\boldsymbol{\psi_V^\pi}(s)^\top \boldsymbol{w}_{\hat{r}})^{1/T}]/[(\sum_{\boldsymbol{\psi'_V} \in \Psi_V} \boldsymbol{\psi'^\pi_V}(s)^\top \boldsymbol{w}_{\hat{r}})^{1/T}]$, where temperature T is a constant hyperparameter. The resulting approximation of $V_{\hat{r}}^*(s)$ is therefore defined as $\tilde{V}_{\hat{r}}^*(s) \triangleq \sum_{\boldsymbol{\psi_V} \in \Psi_V} softmax_{\Psi_V}(\boldsymbol{\psi_V^\pi}(s)^\top \boldsymbol{w}_{\hat{r}})[\boldsymbol{\psi_V^\pi}(s)^\top \boldsymbol{w}_{\hat{r}}]$. Similarly, to approximate $Q_{\hat{r}}^*(s,a)$, $softmax_{\Psi_Q}(\boldsymbol{\psi_Q^\pi}(s,a)^\top \boldsymbol{w}_{\hat{r}}) \triangleq [(\boldsymbol{\psi_Q^\pi}(s,a)^\top \boldsymbol{w}_{\hat{r}})^{1/T}]/[(\sum_{\boldsymbol{\psi'_Q} \in \Psi} \boldsymbol{\psi'^\pi_Q}(s,a)^\top \boldsymbol{w}_{\hat{r}})^{1/T}]$ and $\tilde{Q}_{\hat{r}}^*(s,a) \triangleq \sum_{\boldsymbol{\psi_Q} \in \Psi_Q} softmax_{\Psi_Q}(\boldsymbol{\psi_Q^\pi}(s,a)^\top \boldsymbol{w}_{\hat{r}})[\boldsymbol{\psi_Q^\pi}(s,a)^\top \boldsymbol{w}_{\hat{r}}]$. Consequently, from Eqns. 4 and 5, the corresponding approximation $\tilde{P}_{regret}$ of the regret preference model is:

$$\tilde{P}_{regret}(\sigma_1 \succ \sigma_2|\hat{r}) = logistic\left(\sum_{t=0}^{|\sigma_2|-1}\left[\tilde{V}_{\hat{r}}^*(s_t^{\sigma_2}) - \tilde{Q}_{\hat{r}}^*(s_t^{\sigma_2}, a_t^{\sigma_2})\right] - \sum_{t=0}^{|\sigma_1|-1}\left[\tilde{V}_{\hat{r}}^*(s_t^{\sigma_1}) - \tilde{Q}_{\hat{r}}^*(s_t^{\sigma_1}, a_t^{\sigma_1})\right]\right) \quad (6)$$

The algorithm In Algorithm 1, lines 8–11 describe the supervised-learning optimization using the approximation $\tilde{P}_{regret}$, and the prior lines create Ψ_Q and Ψ_V. Specifically, given a set of input policies (line 1), for each such policy π_{SF}, successor feature functions $\Psi_Q^{\pi_{SF}}$ and $\Psi_V^{\pi_{SF}}$ are estimated (line 4), which by default would be performed by a minor extension of a standard policy evaluation algorithm as detailed by Barreto et al. (2016). Note that the reward function that is ultimately learned is not restricted to be in the input set of reward functions, which is used only to create an approximation of regret.

One potential source of the input set of policies is a set of reward functions, where each input policy is the result of policy improvement on one reward function. We follow this method in our experiments, randomly generating reward functions and then estimating an optimal policy for each reward function. Specifically, for each reward function, we seek the the maximum entropy optimal policy, which resolves ties among optimal actions in a state via a uniform distribution over those optimal actions.

Algorithm 1 Linear reward learning with regret preference model (P_{regret}), using successor features

1: Input: a set of policies
2: $\Psi \leftarrow \varnothing$
3: **for** *each reward function policy* π_{SF} *in the input set* **do**
4: estimate $\psi_Q^{\pi_{SF}}$ and $\psi_V^{\pi_{SF}}$ (if not estimated already during step 4)
5: add $\psi_Q^{\pi_{SF}}$ to Ψ_Q
6: add $\psi_V^{\pi_{SF}}$ to Ψ_V
7: **end for**
8: **repeat**
9: optimize $\boldsymbol{w}_{\hat{r}}$ by loss of Eqn. 1, calculating $\tilde{P}_{regret}(\sigma_1 \succ \sigma_2 | \hat{r})$ via Eqn. 6, using Ψ_Q and Ψ_V
10: **until** *stopping criteria are met*
11: **return** $\boldsymbol{w}_{\hat{r}}$

Further details of our instantiation of Algorithm 1 for the delivery domain can be found in Appendix F.1, along with preliminary guidance for choosing an input set of policies (Appendix F.1.1) and for extending it to reward functions that might be non-linear (Appendix F.1.2).

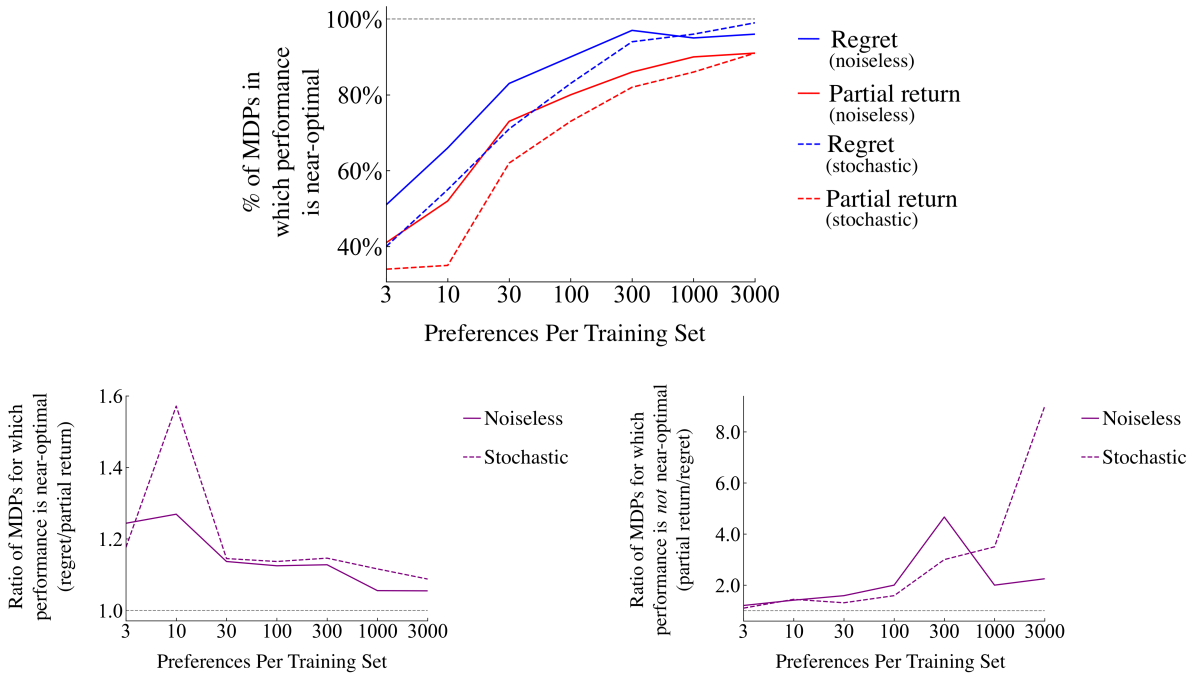

Figure 10: Performance comparison over 100 randomly generated deterministic MDPs when each preference model creates its own training dataset and learns from it. Performance with the regret preference model is consistently better, regardless of training set size or whether preferences are generated stochastically. The bottom-left and bottom-right plots are created from the top plot. The bottom-left plot shows the ratio of between each preference model's success rate. The bottom-right plot shows the ratio between each preference model's rate of failure to reach near-optimal performance. For easier visual comparison, the ratios of each plot are chosen such that higher values indicate better performance by the regret preference model.

6.2 Results from synthetic preferences

Before considering human preferences, we first ask how each preference model performs when it correctly describes how the preferences in its training set were generated. In other words, we investigate empirically how well the preference model could perform if humans perfectly adhered to it. Recall that the ground-truth reward function, r, is used to create these preferences but is inaccessible to the reward-learning algorithms.

For these evaluations, either a stochastic or noiseless preference model acts a preference generator to create a preference dataset. Then the stochastic version of the same model is used for reward learning, which prevents the introduction of a hyperparameter. Note that the stochastic preference model can approach determinism through scaling the reward function, so learning a reward function with the stochastic preference model from deterministically generated preferences does not remove our ability to fit a reward function to those preferences. For the noiseless case, the deterministic preference generator compares a segment pair's $\Sigma_\sigma r$ values for $P_{\Sigma r}$ or their $regret(\sigma|r)$ values for P_{regret}. Note that through reward scaling the preference generators approach determinism in the limit, so this noiseless analysis examines minimal-entropy versions of the two preference-generating models. (The opposite extreme, uniformly random preferences, would remove all information from preferences and therefore is not examined.) In the stochastic case, for each preference model, each segment pair is labeled by sampling from that preference generator's output distribution (Eqs 2 or 5), using the unscaled ground-truth reward function.

We created 100 deterministic MDPs that instantiate variants of our delivery domain (see Section 4.1). To create each MDP, we sampled from sets of possible widths, heights, and reward component values, and the resultant grid cells were randomly populated with a destination, objects, and road surface types (see Appendix F.2 for details). Each segment in the preference datasets for each MDP was generated by choosing a start state and three actions, all uniformly randomly. For a set number of preferences, each method had the same set of segment pairs in its preference dataset. Figure 10 shows the percentage of MDPs in which each preference model results in near-optimal performance. The regret preference model outperforms the partial return model at every dataset size, both with and without noise. By a Wilcoxon paired signed-rank test on normalized mean returns, $p<0.05$ for 86% of these comparisons and $p<0.01$ for 57% of them, as reported in Appendix F.2.

Further analyses can be found in Appendix F.2: with stochastic transitions, with different segment lengths, without segments that terminate before their final transition, and with additional novel preference models.

6.3 Results from human preferences

We now consider the reward-learning performance of each preference model on preferences generated by humans for our specific delivery task. We randomly assign human preferences from our gathered dataset to different numbers of same-sized partitions, resulting in different training set sizes, and test each preference model on each partition. Figure 11 shows the results. With smaller training sets (20–100 partitions), the regret preference model results in near-optimal performance more often. With larger training sets (1–10 partitions), both preference models always reach near-optimal return, but the mean return from the regret preference model is higher for all of these partitions except for only 3 partitions in the 10-partition test. Applying a Wilcoxon paired signed-rank test on normalized mean return to each group with 5 or more partitions, $p<0.05$ for all numbers of partitions except 100 and $p<0.01$ for 20 and 50 partitions. To summarize, we find that both the regret and the partial return preference models achieve near-optimal performance when the dataset is sufficiently large—although the performance of the regret preference model is nonetheless almost always higher—and we also find that regret achieves near-optimal performance more often with smaller datasets.

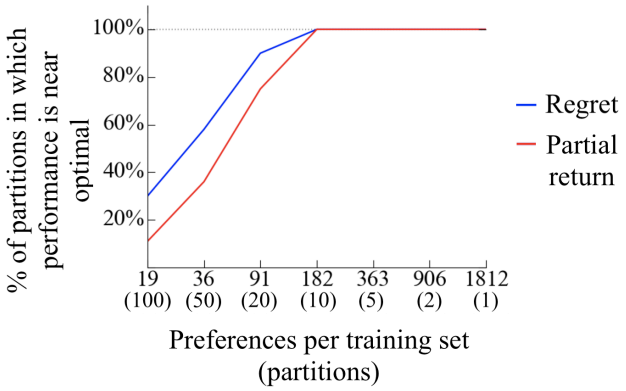

Figure 11: Performance comparison over various amounts of human preferences. Each partition has the number of preferences shown or one less.

Using the human preferences dataset, Appendix F.3 contains further analyses: learning without segments that terminate before their final transition, learning via additional novel preference models, and testing the learned reward functions on other MDPs with the same ground-truth reward function.

7 Conclusion

Over numerous evaluations with human preferences, our proposed regret preference model (P_{regret}) shows improvements summarized below over the previous partial return preference model ($P_{\Sigma r}$). When each preference model generates the preferences for its own infinite and exhaustive training set, we prove that P_{regret} identifies the set of optimal policies, whereas $P_{\Sigma r}$ is not guaranteed to do so in multiple common contexts. With finite training data of synthetic preferences, P_{regret} also empirically results in learned policies that tend to outperform those resulting from $P_{\Sigma r}$. This superior performance of P_{regret} is also seen with human preferences. In summary, our analyses suggest that regret preference models are more effective both descriptively with respect to human preferences and also normatively, as the model we want humans to follow if we had the choice.

Independent of P_{regret}, this paper also reveals that segments' changes in state values provide information about human preferences that is not fully provided by partial return. More generally, we show that the choice of preference model impacts the performance of learned reward functions.

This study motivates several new directions for research. Future work could address any of the limitations detailed in Appendix A.1. Specifically, future work could further test the general superiority of P_{regret} or apply it to deep learning settings. Additionally, *prescriptive* methods could be developed via the subject interface or elsewhere to nudge humans to conform more to P_{regret} or to other normatively appealing preference models. Lastly, this work provided conclusive evidence that the choice of preference model is impactful. Subsequent efforts could seek preference models that are even more effective with preferences from actual humans.

Acknowledgements

References

Pieter Abbeel and Andrew Y Ng. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the twenty-first international conference on Machine learning*, pp. 1, 2004.

Riad Akrour, Marc Schoenauer, and Michele Sebag. Preference-based policy learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 12–27. Springer, 2011.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016.

Saurabh Arora and Prashant Doshi. A survey of inverse reinforcement learning: Challenges, methods and progress. *Artificial Intelligence*, 297:103500, 2021.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

André Barreto, Will Dabney, Rémi Munos, Jonathan J Hunt, Tom Schaul, Hado Van Hasselt, and David Silver. Successor features for transfer in reinforcement learning. *arXiv preprint arXiv:1606.05312*, 2016.

Marc G Bellemare, Salvatore Candido, Pablo Samuel Castro, Jun Gong, Marlos C Machado, Subhodeep Moitra, Sameera S Ponda, and Ziyu Wang. Autonomous navigation of stratospheric balloons using reinforcement learning. *Nature*, 588(7836):77–82, 2020.

Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, Przemysław Dębiak, Christy Dennison, David Farhi, Quirin Fischer, Shariq Hashme, Chris Hesse, et al. Dota 2 with large scale deep reinforcement learning. *arXiv preprint arXiv:1912.06680*, 2019.

Erdem Bıyık, Dylan P Losey, Malayandi Palan, Nicholas C Landolfi, Gleb Shevchuk, and Dorsa Sadigh. Learning reward functions from diverse sources of human feedback: Optimally integrating demonstrations and preferences. *The International Journal of Robotics Research*, pp. 02783649211041652, 2021.

Daniel Brown, Russell Coleman, Ravi Srinivasan, and Scott Niekum. Safe imitation learning via fast bayesian reward inference from preferences. In *International Conference on Machine Learning*, pp. 1165–1177. PMLR, 2020.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 4299–4307, 2017.

Yuchen Cui, Qiping Zhang, Alessandro Allievi, Peter Stone, Scott Niekum, and W Bradley Knox. The EMPATHIC framework for task learning from implicit human feedback. *arXiv preprint arXiv:2009.13649*, 2020.

Jonas Degrave, Federico Felici, Jonas Buchli, Michael Neunert, Brendan Tracey, Francesco Carpanese, Timo Ewalds, Roland Hafner, Abbas Abdolmaleki, Diego de Las Casas, et al. Magnetic control of tokamak plasmas through deep reinforcement learning. *Nature*, 602(7897):414–419, 2022.

William Fedus, Carles Gelada, Yoshua Bengio, Marc G Bellemare, and Hugo Larochelle. Hyperbolic discounting and learning over multiple horizons. *arXiv preprint arXiv:1902.06865*, 2019.

Shane Frederick, George Loewenstein, and Ted O'donoghue. Time discounting and time preference: A critical review. *Journal of economic literature*, 40(2):351–401, 2002.

Amelia Glaese, Nat McAleese, Maja Trębacz, John Aslanides, Vlad Firoiu, Timo Ewalds, Maribeth Rauh, Laura Weidinger, Martin Chadwick, Phoebe Thacker, et al. Improving alignment of dialogue agents via targeted human judgements. *arXiv preprint arXiv:2209.14375*, 2022.

Adam Gleave, Mohammad Taufeeque, Juan Rocamonde, Erik Jenner, Steven H. Wang, Sam Toyer, Maximilian Ernestus, Nora Belrose, Scott Emmons, and Stuart Russell. imitation: Clean imitation learning implementations. arXiv:2211.11972v1 [cs.LG], 2022. URL https://arxiv.org/abs/2211.11972.

Joey Hejna and Dorsa Sadigh. Inverse preference learning: Preference-based rl without a reward function. *arXiv preprint arXiv:2305.15363*, 2023.

Borja Ibarz, Jan Leike, Tobias Pohlen, Geoffrey Irving, Shane Legg, and Dario Amodei. Reward learning from human preferences and demonstrations in atari. *arXiv preprint arXiv:1811.06521*, 2018.

Kuno Kim, Shivam Garg, Kirankumar Shiragur, and Stefano Ermon. Reward identification in inverse reinforcement learning. In *International Conference on Machine Learning*, pp. 5496–5505. PMLR, 2021.

W Bradley Knox, Alessandro Allievi, Holger Banzhaf, Felix Schmitt, and Peter Stone. Reward (mis)design for autonomous driving. *arXiv preprint arXiv:2104.13906*, 2021.

Zeb Kurth-Nelson and A David Redish. Temporal-difference reinforcement learning with distributed representations. *PLoS One*, 4(10):e7362, 2009.

Kimin Lee, Laura Smith, and Pieter Abbeel. Pebble: Feedback-efficient interactive reinforcement learning via relabeling experience and unsupervised pre-training. *arXiv preprint arXiv:2106.05091*, 2021a.

Kimin Lee, Laura Smith, Anca Dragan, and Pieter Abbeel. B-pref: Benchmarking preference-based reinforcement learning. *arXiv preprint arXiv:2111.03026*, 2021b.

R Duncan Luce. *Individual choice behavior: A theoretical analysis*. John Wiley, 1959.

Andrew Y Ng and Stuart Russell. Algorithms for inverse reinforcement learning. In *Seventeenth International Conference on Machine Learning (ICML)*, 2000.

OpenAI. Chatgpt: Optimizing language models for dialogue. OpenAI Blog https://openai.com/blog/chatgpt/, 2022. Accessed: 2022-12-20.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*, 2022.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. Curran Associates, Inc., 2019. URL http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

A David Redish and Zeb Kurth-Nelson. Neural models of delay discounting. In *Impulsivity: The behavioral and neurological science of discounting.*, pp. 123–158. American Psychological Association, 2010.

Stuart Russell and Peter Norvig. *Artificial intelligence: a modern approach*. Pearson, 2020.

Dorsa Sadigh, Anca D Dragan, Shankar Sastry, and Sanjit A Seshia. Active preference-based learning of reward functions. *Robotics: Science and Systems*, 2017.

Andrew W Senior, Richard Evans, John Jumper, James Kirkpatrick, Laurent Sifre, Tim Green, Chongli Qin, Augustin Žídek, Alexander WR Nelson, Alex Bridgland, et al. Improved protein structure prediction using potentials from deep learning. *Nature*, 577(7792):706–710, 2020.

Roger N Shepard. Stimulus and response generalization: A stochastic model relating generalization to distance in psychological space. *Psychometrika*, 22(4):325–345, 1957.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

Joar Skalse, Matthew Farrugia-Roberts, Stuart Russell, Alessandro Abate, and Adam Gleave. Invariance in policy optimisation and partial identifiability in reward learning. *arXiv preprint arXiv:2203.07475*, 2022.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature*, 575(7782):350–354, 2019.

John Von Neumann and Oskar Morgenstern. *Theory of games and economic behavior*. Princeton university press, 1944.

Xiaofei Wang, Kimin Lee, Kourosh Hakhamaneshi, Pieter Abbeel, and Michael Laskin. Skill preferences: Learning to extract and execute robotic skills from human feedback. In *Conference on Robot Learning*, pp. 1259–1268. PMLR, 2022.

Peter R. Wurman, Samuel Barrett, Kenta Kawamoto, James MacGlashan, Kaushik Subramanian, Thomas J. Walsh, Roberto Capobianco, Alisa Devlic, Franziska Eckert, Florian Fuchs, Leilani Gilpin, Varun Kompella, Piyush Khandelwal, HaoChih Lin, Patrick MacAlpine, Declan Oller, Craig Sherstan, Takuma Seno, Michael D. Thomure, Houmehr Aghabozorgi, Leon Barrett, Rory Douglas, Dion Whitehead, Peter Duerr, Peter Stone, Michael Spranger, , and Hiroaki Kitano. Outracing champion gran turismo drivers with deep reinforcement learning. *Nature*, 62:223–28, Feb. 2022. doi: 10.1038/s41586-021-04357-7.

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.