Occam's razor is insufficient to infer the preferences of irrational agents

Stuart Armstrong * †
Future of Humanity Institute
University of Oxford
stuart.armstrong@philosophy.ox.ac.uk

Sören Mindermann* ‡
Vector Institute
University of Toronto
soeren.mindermann@gmail.com

Abstract

Inverse reinforcement learning (IRL) attempts to infer human rewards or preferences from observed behavior. Since human planning systematically deviates from rationality, several approaches have been tried to account for specific human shortcomings. However, the general problem of inferring the reward function of an agent of unknown rationality has received little attention. Unlike the well-known ambiguity problems in IRL, this one is practically relevant but cannot be resolved by observing the agent's policy in enough environments. This paper shows (1) that a No Free Lunch result implies it is impossible to uniquely decompose a policy into a planning algorithm and reward function, and (2) that even with a reasonable simplicity prior/Occam's razor on the set of decompositions, we cannot distinguish between the true decomposition and others that lead to high regret. To address this, we need simple 'normative' assumptions, which cannot be deduced exclusively from observations.

1 Introduction

In today's reinforcement learning systems, a simple reward function is often hand-crafted, and still sometimes leads to undesired behaviors on the part of RL agent, as the reward function is not well aligned with the operator's true goals[4]. As AI systems become more powerful and autonomous, these failures will become more frequent and grave as RL agents exceed human performance, operate at time-scales that forbid constant oversight, and are given increasingly complex tasks — from driving cars to planning cities to eventually evaluating policies or helping run companies. Ensuring that the agents behave in alignment with human values is known, appropriately, as the *value alignment problem* [Amodei et al., 2016, Hadfield-Menell et al., 2016, Russell et al., 2015, Bostrom, 2014, Leike et al., 2017].

One way of resolving this problem is to infer the correct reward function by observing human behaviour. This is known as Inverse reinforcement learning (IRL) [Ng and Russell, 2000, Abbeel and Ng, 2004, Ziebart et al., 2008]. Often, learning a reward function is preferred over imitating a policy: when the agent must outperform humans, transfer to new environments, or be interpretable. The reward function is also usually a (much) more succinct and robust task representation than the policy, especially in planning tasks [Abbeel and Ng, 2004]. Moreover, supervised learning of long-range and goal-directed behavior is often difficult without the reward function [Ratliff et al., 2006].

*Equal contribution.

†Further affiliation: Machine Intelligence Research Institute, Berkeley, USA.

‡Work performed at Future of Humanity Institute.

[4]See for example the game CoastRunners, where an RL agent didn't finish the course, but instead found a bug allowing it to get a high score by crashing round in circles `https://blog.openai.com/faulty-reward-functions/`.

32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada.

Usually, the reward function is inferred based on the assumption that human behavior is optimal or noisily optimal. However, it is well-known that humans deviate from rationality in *systematic*, non-random ways [Tversky and Kahneman, 1975]. This can be due to specific biases such as time-inconsistency, loss aversion and hundreds of others, but also limited cognitive capacity, which leads to forgetfulness, limited planning and false beliefs. This limits the use of IRL methods for tasks that humans don't find trivial.

Some IRL approaches address specific biases [Evans et al., 2015b,a], and others assume noisy rationality [Ziebart et al., 2008, Boularias et al., 2011]. But a general framework for inferring the reward function from suboptimal behavior does not exist to our knowledge. Such a framework needs to infer two unobserved variables simultaneously: the human reward function and their planning algorithm[5] which connects the reward function with behaviour, henceforth called a *planner*.

The task of observing human behaviour (or the human policy) and inferring from it the human reward function and planner will be termed *decomposing* the human policy. This paper will show there is a No Free Lunch theorem in this area: it is impossible to get a unique decomposition of human policy and hence get a unique human reward function. Indeed, *any* reward function is possible. And hence, if an IRL agent acts on what it believes is the human policy, the potential regret is near-maximal. This is another form of unidentifiability of the reward function, beyond the well-known ones [Ng and Russell, 2000, Amin and Singh, 2016].

The main result of this paper is that, unlike other No Free Lunch theorems, this unidentifiability does not disappear when regularising with a general simplicity prior that formalizes Occam's razor [Vitanyi and Li, 1997]. This result will be shown in two steps: first, that the simplest decompositions include degenerate ones, and secondly, that the most 'reasonable' decompositions according to human judgement are of high complexity.

So, although current IRL methods can perform well on many well-specified problems, they are fundamentally and philosophically incapable of establishing a 'reasonable' reward function for the human, no matter how powerful they become. In order to do this, they will need to build in 'normative assumptions': key assumptions about the reward function and/or planner, that cannot be deduced from observations, and allow the algorithm to focus on good ways of decomposing the human policy.

Future work will sketch out some potential normative assumptions that can be used in this area, making use of the fact that humans assess each other to be irrational, and often these assessments agree. In view of the No Free Lunch result, this shows that humans must share normative assumptions.

One of these 'normative assumption' approaches is briefly illustrated in an appendix, while another appendix demonstrates how to use the planner-reward formalism to define when an agent might be manipulating or overriding human preferences. This happens when the agent pushes the human towards situations where their policy is very suboptimal according to their reward function.

2 Related Work

In the first IRL papers from Ng and Russell [2000] and Abbeel and Ng [2004] a max-margin algorithm was used to find the reward function under which the observed policy most outperforms other policies. Suboptimal behavior was first addressed explicitly by Ratliff et al. [2006] who added slack variables to allow for suboptimal behavior. This finds reward functions such that the observed policy outperforms most other policies and the biggest margin by which another policy outperforms it is minimal, i.e. the observed policy has low regret. Shiarlis et al. [2017] introduce a modern max-margin technique with an approximate planner in the optimisation.

However, the max-margin approach has mostly been replaced by the max entropy IRL [Ziebart et al., 2008]. Here, the assumption is that observed actions or trajectories are chosen with probability proportional to the exponent of their value. This assumes a specific suboptimal planning algorithm which is *noisily* rational (also known as *Boltzmann*-rational). Noisy rationality explains human behavior on various data sets better [Hula et al., 2015]. However, Evans et al. [2015b] and Evans et al. [2015a] showed that this can fail since humans deviate from rationality in systematic, non-random ways. If noisy rationality is assumed, repeated suboptimal actions throw off the inference.

[5] Technically we only need to infer the human reward function, but inferring that from behaviour requires some knowledge of the planning algorithm.

Literature on inferring the reasoning capabilities of an agent is scarce. Evans et al. [2015b] and Evans et al. [2015a] use Bayesian inference to identify specific planning biases such as myopic planning and hyperbolic time-discounting. They simultaneously infer the agent's preferences. Cundy and Filan [2018] adds bias resulting from hierarchical planning. Hula et al. [2015] similarly let agents infer features of their opponent's reasoning such as planning depth and impulsivity in simple economic games. Recent work learns the planning algorithm with two assumptions: being close to noisily rational in a high-dimensional planner space and supervised planner-learning [Anonymous, 2019].

The related ideas of meta-reasoning [Russell, 2016], computational rationality [Lewis et al., 2014] and resource rationality [Griffiths et al., 2015] may create the possibility to redefine irrational behavior as rational in an 'ancestral' distribution of environments where the agent optimises its rewards by choosing among the limited computations it is able to perform or jointly minimising the cost of computation and maximising reward. This could in theory redefine many biases as computationally optimal in some distribution of environments and provide priors on human planning algorithms. Unfortunately the problem of doing this in practice seems to be extremely difficult — and it assumes that human goals are roughly the same as evolution's goals, which is certainly not the case.

3 Problem setup and background

A human will be performing a series of actions, and from these, an agent will attempt to estimate both the human's reward function and their planning algorithm.

The environment M in which the human operates is an MDP/R, a Markov Decision Process without reward function (a *world-model* [Hadfield-Menell et al., 2017]). An MDP/R is defined as a tuple, $\langle \mathcal{S}, \mathcal{A}, T, \hat{s} \rangle$ consisting of a discrete state space $\mathcal{S}$, a finite action space $\mathcal{A}$, a fixed starting state $\hat{s}$, and a probabilistic transition function $T : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ to the next state (also called the *dynamics*). At each step, the human is in a certain state s, takes a certain action a, and ends up in a new state s' as given by $T(s' \mid s, a)$.

Let $\mathcal{R} = \{R : \mathcal{S} \times \mathcal{A} \to [-1, 1]\} = [-1, 1]^{\mathcal{S} \times \mathcal{A}}$ be the space of candidate reward functions; a given R will map any state-reward pair to a reward value in the interval $[-1, 1]$.

Let Π be the space of deterministic, Markovian policies. So Π is the space of functions $\mathcal{S} \to \mathcal{A}$. The human will be following the policy $\dot{\pi} \in \Pi$.

The results of this paper apply to both discounted rewards and episodic environments settings[6].

3.1 Planners and reward functions: decomposing the policy

The human has their reward function, and then follows a policy that presumably attempts to maximise it. Therefore there is something that bridges between the reward function and the policy: a piece of greater or lesser rationality that transforms knowledge of the reward function into a plan of action.

This bridge will be modeled as a *planner* $p : \mathcal{R} \to \Pi$, a function that takes a reward and outputs a policy. This planner encodes all the rationality, irrationality, and biases of the human. Let $\mathcal{P}$ be the set of planners. The human is therefore defined by a *planner-reward* pair $(p, R) \in \mathcal{P} \times \mathcal{R}$. Similarly, (p, R) with $p(R) = \pi$ is a *decomposition* of the policy π. The task of the agent is to find a 'good' decomposition of the human policy $\dot{\pi}$.

3.2 Compatible pairs and evidence

The agent can observe the human's behaviour and infer their policy from that. In order to simplify the problem and separate out the effect of the agent's learning, we will assume the agent has perfect knowledge of the human policy $\dot{\pi}$ and of the environment M. At this point, the agent cannot learn anything by observing the human's actions, as it can already perfectly predict these.

Then a pair (p, R) is defined to be *compatible* with $\dot{\pi}$, if $p(R) = \dot{\pi}$ — thus that pair is a possible candidate for decomposing the human policy into the human's planner and reward function.

[6]The setting is only chosen for notational convenience: it also emulates discrete POMDPs, non-Markovianness (eg by encoding the whole history in the state) and pseudo-random policies.

4 Irrationality-based unidentifiability

Unidentifiability of the reward is a well-known problem in IRL [Ng and Russell, 2000]. Amin and Singh [2016] categorise the problem into *representational* and *experimental* unidentifiability. The former means that adding a constant to a reward function or multiplying it with a positive scalar does not change what is optimal behavior. This is unproblematic as rescaling the reward function doesn't change the preference ordering. The latter can be resolved by observing optimal policies in a whole class of MDPs which contains all possible transition dynamics. We complete this framework with a third kind of identifiability, which arises when we observe suboptimal agents. This kind of unidentifiability is worse as it cannot necessarily be resolved by observing the agent in many tasks. In fact, it can lead to almost arbitrary regret.

4.1 Weak No Free Lunch: unidentifiable reward function and half-maximal regret

The results in this section show that without assumptions about the rationality of the human, all attempts to optimise their reward function are essentially futile. Everitt et al. [2017] work in a similar setting as we do: in their case, a corrupted version of the reward function is observed. The problem our case is that a 'corrupted' version $\dot{\pi}$ of an optimal policy $\pi^*_{\dot{R}}$ is observed and used as information to optimise for the ideal reward $\dot{R}$. A No Free Lunch result analogous to theirs applies in our case; both resemble the No Free Lunch theorems for optimisation [Wolpert and Macready, 1997].

More philosophically, this result is as an instance of the well-known *is-ought* problem from meta-ethics. Hume [1888] argued that what *ought* to be (here, the human's reward function) can never be concluded from what *is* (here, behavior) without extra assumptions. Equivalently, the human reward function cannot be inferred from behavior without assumptions about the planning algorithm p. In probabilistic terms, the likelihood $P(\pi|R) = \sum_{p\in\mathcal{P}} P(\pi \mid R, p)P(p)$ is undefined without $P(p)$. As shown in Section 5 and Section 5.2, even a simplicity prior on p and R will not help.

4.1.1 Unidentifiable reward functions

Firstly, we note that compatibility ($p(R) = \dot{\pi}$), puts no restriction on R, and few restrictions on p:

Theorem 1. *For all $\pi \in \Pi$ and $R \in \mathcal{R}$, there exists a $p \in \mathcal{P}$ such that $p(R) = \pi$.*

For all $p \in \mathcal{P}$ and $\pi \in \Pi$ in the image of p, there exists an R such that $p(R) = \pi$.

Proof. Trivial proof: define the planner[7] p as mapping all of $\mathcal{R}$ to π; then $p(R) = \pi$. The second statement is even more trivial, as π is in the image of p, so there must exist R with $p(R) = \pi$. □

4.1.2 Half-maximal regret

The above shows that the reward function cannot be constrained by observation of the human, but what about the expected long-term value? Suppose that an agent is unsure what the actual human reward function is; if the agent itself is acting in an MDP/R, can it follow a policy that minimises the possible downside of its ignorance?

This is prevented by a recent No Free Lunch theorem. Being ignorant of the reward function one should maximise is equivalent of having a *corrupted reward channel* with arbitrary corruption. In that case, Everitt et al. [2017] demonstrated that whatever policy π the agent follows, there is a $R \in \mathcal{R}$ for which π is half as bad as the worst policy the agent could have followed. Specifically, let $V^\pi_R(s)$ be the expected return of reward function R from state s, given that the agent follows policy π. If π was the optimal policy for R, then this can be written as $V^*_R(s)$. The regret of π for R at s is given by the difference:

$$\mathrm{Reg}(\pi, R)(s) = V^*_R(s) - V^\pi_R(s).$$

Then Everitt et al. [2017] demonstrates that for any π,

$$\max_{R\in\mathcal{R}} \mathrm{Reg}(\pi, R)(s) \geq \frac{1}{2}\left(\max_{\pi'\in\Pi, R\in\mathcal{R}} \mathrm{Reg}(\pi', R)(s)\right).$$

So for any compatible $(p, R) = \dot{\pi}$, we cannot rule out that maximizing R leads to at least half of the worst-case regret.

[7] This is the 'indifferent' planner p_π of subsubsection 5.1.1.

5 Simplicity of degenerate decompositions

Like many No Free Lunch theorems, the result of the previous section is not surprising given there are no assumptions about the planning algorithm. No Free Lunch results are generally avoided by placing a simplicity prior on the algorithm, dataset, function class or other object [Everitt et al., 2014]. This amounts to saying algorithms can benefit from regularisation. This section is dedicated to showing that, surprisingly, simplicity does not solve the No Free Lunch result.

Our simplicity measure is minimum description length of an object, defined as Kolmogorov complexity [Kolmogorov, 1965], the length of the shortest program that outputs a string describing the object. This is the most general formalization of Occam's razor we know of [Vitanyi and Li, 1997]. Appendix A explores how the results extend to other measures of complexity, such as those that include computation time. We start with informal versions of our main results.

Theorem 2 (Informal simplicity theorem)**.** *Let $(\dot{p}, \dot{R})$ be a 'reasonable' planner-reward pair that captures our judgements about the biases and rationality of a human with policy $\dot{\pi} = \dot{p}(\dot{R})$. Then there are degenerate planner-reward pairs, compatible with $\dot{\pi}$, of lower complexity than $(\dot{p}, \dot{R})$, and a pair $(\dot{p}', -\dot{R})$ of similar complexity to $(\dot{p}, \dot{R})$, but with opposite reward function.*

There are a few issues with this theorem as it stands. Firstly, simplicity in algorithmic information theory is relative to the computer language (or equivalently Universal Turing Machine) L used [Ming and Vitányi, 2014, Calude, 2002], and there exists languages in which the theorem is clearly false: one could choose a degenerate language in which $(\dot{p}, \dot{R})$ is encoded by the string '0', for example, and all other planner-reward pairs are of extremely long length. What constitutes a 'reasonable' language is a long-standing open problem, see Leike et al. [2017] and Müller [2010]. For any pair of languages, complexities differ only by a constant, the amount required for one language to describe the other, but this constant can be arbitrarily large.

Nevertheless, this section will provide grounds for the following two semi-formal results:

Proposition 3. *If $\dot{\pi}$ is a human policy, and L is a 'reasonable' computer language, then there exists degenerate planner-reward pairs amongst the pairs of lowest complexity compatible with $\dot{\pi}$.*

Proposition 4. *If $\dot{\pi}$ is a human policy, and L is a 'reasonable' computer language with $(\dot{p}, \dot{R})$ a compatible planner-reward pair, then there exist a pair $(\dot{p}', -\dot{R})$ of comparable complexity to $(\dot{p}, \dot{R})$, but opposite reward function.*

The last part of Theorem 2, the fact that any 'reasonable' $(\dot{p}, \dot{R})$ is expected to be of higher complexity, will be addressed in Section 6.

5.1 Simple degenerate pairs

The argument in this subsection will be that 1) the complexity of $\dot{\pi}$ is close to a lower bound on any pair compatible with it and 2) degenerate decompositions are themselves close to this bound. The first statement follows because for any decomposition (p, R) compatible with $\dot{\pi}$, the map $(p, R) \mapsto p(R) = \dot{\pi}$ will be a simple one, adding little complexity. And if a compatible pair (p', R') can be from $\dot{\pi}$ with little extra complexity, then it too will have a complexity close to the minimal complexity of any other pair compatible with it. Therefore we will first produce three degenerate pairs that can be simply constructed from $\dot{\pi}$.

5.1.1 The degenerate pairs

We can define the trivial constant reward function 0, and the greedy planner p_g. The greedy planner p_g acts by taking the action that maximises the immediate reward in the current state and the next action. Thus[8] $p_g(R)(s) = \text{argmax}_a R(s, a)$. We can also define the anti-greedy planner $-p_g$, with $-p_g(R)(s) = \text{argmin}_a R(s, a)$. In general, it will be useful to define the negative of a planner:

Definition 5. If $p : \mathcal{R} \to \Pi$ is a planner, the planner $-p$ is defined by $-p(R) = p(-R)$.

For any given policy π, we can define the *indifferent* planner p_π, which maps any reward function to π. We can also define the reward function R_π, so that $R_\pi(s, a) = 1$ if $\pi(s) = a$, and $R_\pi(s, a) = 0$ otherwise. The reward function $-R_\pi$ is defined to be the negative of R_π. Then:

[8] Recall that p_g is a planner, $p_g(R)$ is a policy, so $p_g(R)$ can be applied to states, and $p_g(R)(s)$ is an action.

Lemma 6. *The pairs* $(p_\pi, 0)$*,* (p_g, R_π)*, and* $(-p_g, -R_\pi)$ *are all compatible with* π*.*

Proof. Since the image p_π is π, $p_\pi(0) = \pi$. Now, $R_\pi(s, a) > 0$ iff $\pi(s) = a$, hence for all s:

$$p_g(R_\pi)(s) = \operatorname*{argmax}_a R_\pi(s, a) = \pi(s),$$

so $p_g(R_\pi) = \pi$. Then $-p_g(-R_\pi) = p_g(-(-R_\pi)) = p_g(R_\pi) = \pi$, by Definition 5. □

5.1.2 Complexity of basic operations

We will look the operations that build the degenerate planner-reward pairs from any compatible pair:

1. For any planner p, $f_1(p) = (p, 0)$ as a planner-reward pair.
2. For any reward function R, $f_2(R) = (p_g, R)$.
3. For any planner-reward pair (p, R), $f_3(p, R) = p(R)$.
4. For any planner-reward pair (p, R), $f_4(p, R) = (-p, -R)$.
5. For any policy π, $f_5(\pi) = p_\pi$.
6. For any policy π, $f_6(\pi) = R_\pi$.

These will be called the basic operations, and there are strong arguments that reasonable computer languages should be able to express them with short programs. The operation f_1, for instance, is simply appending the flat trivial 0, f_2 appends a planner defined by the simple[9] search operator argmax, f_3 applies a planner to the object — a reward function — that the planner naturally acts on, f_4 is a double negation, while f_5 and f_6 are simply described in subsubsection 5.1.1.

From these basic operations, we can define three composite operations that map any compatible planner-reward pair to one of the degenerate pairs (the element $F_4 = f_4$ is useful for later definitions). Thus define

$$F = \{F_1 = f_1 \circ f_5 \circ f_3,\ \ F_2 = f_2 \circ f_6 \circ f_3,\ \ F_3 = f_4 \circ f_2 \circ f_6 \circ f_3,\ \ F_4 = f_4\}.$$

For any $\dot{\pi}$-compatible pair (p, R) we have $F_1(p, R) = (p_{\dot{\pi}}, 0)$, $F_2(p, R) = (p_g, R_{\dot{\pi}})$, and $F_3(p, R) = (-p_g, -R_{\dot{\pi}})$ (see the proof of Proposition 7).

Let K_L denote Kolmogorov complexity in the language L: the shortest algorithm in L that generates a particular object. We define the F-complexity of L as

$$\max_{(p,R), F_i \in F} K_L(F_i(p, R)) - K_L(p, R).$$

Thus the F-complexity of L is how much the F_i potentially increase[10] the complexity of pairs.

For a constant $c \geq 0$, this allows us to formalise what we mean by L being a c-reasonable language for F: that the F-complexity of L is at most c. A reasonable language is a c-reasonable language for a c that we feel is intuitively low enough.

5.1.3 Low complexity of degenerate planner-reward pairs

To formalise the concepts 'of lowest complexity', and 'of comparable complexity', choose a constant $c \geq 0$, then (p, R) and (p', R') are of 'comparable complexity' if

$$||K_L(p, R) - K_L(p', R')|| \leq c.$$

For a set $S \subset \mathcal{P} \times \mathcal{R}$, the pair $(p, R) \in S$ is amongst the lowest complexity in S if

$$||K_L(p, R) - \min_{(p', R') \in S} K_L(p', R')|| \leq c,$$

thus K_L is within distance c of the minimum complexity element of S. Now formalize Proposition 3:

[9] In most standard computer languages, argmax just requires a for-loop, a reference to R, a comparison with a previously stored value, and possibly the storage of a new value and the current action.

[10] F-complexity is non-negative: $F_4 \circ F_4$ is the identity, so that $K_L(F_4(p, R)) - K_L(p, R) = -(K_L(F_4(F_4(p, R)) - K_L(F_4(p, R))$, meaning that $\max_{(p,R), F_4} K_L(F_4(p, R)) - K_F(p, R)$ must be non-negative; this is a reason to include F_4 in the definition of F.

Proposition 7. *If $\dot{\pi}$ is the human policy, c defines a reasonable measure of comparable complexity, and L is a c-reasonable language for F, then the degenerate planner-reward pairs $(p_{\dot{\pi}}, 0)$, $(p_g, R_{\dot{\pi}})$, and $(-p_g, -R_{\dot{\pi}})$ are amongst the pairs of lowest complexity among the pairs compatible with $\dot{\pi}$.*

Proof. By Lemma 6, $(p_{\dot{\pi}}, 0)$, $(p_g, R_{\dot{\pi}})$, and $(-p_g, -R_{\dot{\pi}})$ are compatible with $\dot{\pi}$. By the definitions of the f_i and F_i, for(p, R) compatible with $\dot{\pi}$, $f_3((p, R)) = p(R) = \dot{\pi}$ and hence

$$F_1(p, R) = f_1 \circ f_5(\dot{\pi}) = f_1(p_{\dot{\pi}}) = (p_{\dot{\pi}}, 0),$$
$$F_2(p, R) = f_2 \circ f_6(\dot{\pi}) = f_2(R_{\dot{\pi}}) = (p_g, R_{\dot{\pi}}),$$
$$F_3(p, R) = f_4 \circ F_2(p, R) = (-p_g, -R_{\dot{\pi}}).$$

Now pick (p, R) to be the simplest pair compatible with $\dot{\pi}$. Since L is c-reasonable for F, $K_L(p_{\dot{\pi}}, 0) \leq c + K_L(p, R)$. Hence $(p_{\dot{\pi}}, 0)$ is of lowest complexity among the pairs compatible with $\dot{\pi}$; the same argument applies for the other two degenerate pairs. □

5.2 Negative reward

If $(\dot{p}, \dot{R})$ is compatible with $\dot{\pi}$, then so is $(-\dot{p}, -\dot{R}) = f_4(\dot{p}, \dot{R}) = F_4(\dot{p}, \dot{R})$. This immediately implies the formalisation of Proposition 4:

Proposition 8. *If $\dot{\pi}$ is a human policy, c defines a reasonable measure of comparable complexity, L is a c-reasonable language for F, and $(\dot{p}, \dot{R})$ is compatible with $\dot{\pi}$, then $(-\dot{p}, -\dot{R})$ is of comparable complexity to $(\dot{p}, \dot{R})$.*

So complexity fails to distinguish between a reasonable human reward function and its negative.

6 The high complexity of the genuine human reward function

Section 5 demonstrated that there are degenerate planner-reward pairs close to the minimum complexity among all pairs compatible with $\dot{\pi}$. This section will argue that any reasonable pair $(\dot{p}, \dot{R})$ is unlikely to be close to this minimum, and is therefore of higher complexity than the degenerate pairs. Unlike simplicity, reasonable decomposition cannot easily be formalised. Indeed, a formalization would likely already solve the problem, yielding an algorithm to maximize it. Therefore, the arguments in this section are mostly qualitative.

We use reasonable to mean 'compatible with human judgements about rationality'. Since we do not have direct access to such a decomposition, the complexity argument will be about showing the complexity of these human judgements. This argument will proceed in three stages:

1. Any reasonable $(\dot{p}, \dot{R})$ is of high complexity, higher than it may intuitively seem to us.
2. Even given $\dot{\pi}$, any reasonable $(\dot{p}, \dot{R})$ involves a high number of contingent choices. Hence any given $(\dot{p}, \dot{R})$ has high information (and thus high complexity), even given $\dot{\pi}$.
3. Past failures to find a simple $(\dot{p}, \dot{R})$ derived from $\dot{\pi}$ are evidence that this is tricky.

6.1 The complexity of human (ir)rationality

Humans make noisy and biased decisions all the time. Though noise is important [Kahneman et al., 2016], many biases, such as anchoring bias, overconfidence, planning fallacies, and so on, affect humans in a highly systematic way; see Kahneman and Egan [2011] for many examples.

Many people may feel that they have a good understanding of rationality, and therefore assume that assessing the (ir)rationality of any particular decision is not a complicated process. But an intuition for bias does not translate into a process for establishing a $(\dot{p}, \dot{R})$.

Consider the anchoring bias defined in Ariely et al. [2004], where irrelevant information — the last digits of social security numbers — changed how much people were willing to pay for goods. When defining a reasonable $(\dot{p}, \dot{R})$, it does not suffice to be aware of the existence of anchoring bias[11], but

[11] The fact that many cognitive biases have only been discovered recently argue against people having a good intuitive grasp of bias and rationality, as do people's persistent bias blind spots [Scopelliti et al., 2015].

one has to precisely quantify the extent of the bias — why does anchoring bias seem to be stronger for chocolate than for wine, for instance? And why these precise percentages and correlations, and not others? And can people's judgment tell which people are more or less susceptible to anchoring bias? And can one quantify the bias for a single individual, rather than over a sample?

Any given $(\dot{p}, \dot{R})$ can quantify the form and extent of these biases by computing objects like the regret function $\mathrm{Reg}(\dot{p}, \dot{R})(s) := \mathrm{Reg}(\dot{p}(\dot{R}), \dot{R})(s) = V^*_{\dot{R}}(s) - V^{\dot{p}(\dot{R})}_{\dot{R}}(s)$, which measures the divergence between the expected value of the actual and optimal human policies[12]. Thus any given $(\dot{p}, \dot{R})$ — which contains the information to compute quantities like $\mathrm{Reg}(\dot{p}, \dot{R})(s)$ or similar measures of bias[13], in every state — carries a high amount of numerical information about bias, and hence a high complexity.

Since humans do not easily have access to this information, this implies that human judgement of irrationality is subject to Moravec's paradox [Moravec, 1988]. It is similar to, for example, social skills: though it seems intuitively simple to us, it is highly complex to define in algorithmic terms.

Other authors have argued directly for the complexity of human values, from fields as diverse as computer science, philosophy, neuroscience, and economics [Minsky, 1984, Bostrom, 2014, Glimcher et al., 2009, Muehlhauser and Helm, 2012, Yudkowsky, 2011].

6.2 The contingency of human judgement

The previous section showed that reasonable $(\dot{p}, \dot{R})$ carry large amounts of information/complexity, but the key question is whether it requires information *additional* to that in $\dot{\pi}$. This section will show that even when $\dot{\pi}$ is known, there are many contingent choices that need to be made to define any specific reasonable $(\dot{p}, \dot{R})$. Hence any given $(\dot{p}, \dot{R})$ contains a large amount of information beyond that in $\dot{\pi}$, and hence is of higher complexity.

Reasons to believe that human judgement about reasonable $(\dot{p}, \dot{R})$ contains many contingent choices:

- There is a variability of human judgement between cultures. When Miller [1984] compared American and Indian assessments of the same behaviours, they found systematically different explanations for them[14] Basic intuitions about rationality also vary between cultures [Nisbett et al., 2001, Brück, 1999].
- There is a variability of human judgement within a single culture. When Slovic and Tversky [1974] analysed the "Allais Paradox", they found that different people gave different answers as to what the rational behaviour was in their experiments.
- There is evidence of variability of human judgement within the same person. Slovic and Tversky [1974] further attempted to argue for the rationality of one of the answers. This sometimes resulted in the participant sometimes changing their minds, and contradicting their previous assessment of rationality.
- There is a variability of human judgement for the same person assessing their own values, caused by differences as trivial as question ordering [Schuman and Ludwig, 1983]. So human meta-judgement, of own values and rationality, is also contingent and variable.
- People have partial bias blind spots around their own biases [Scopelliti et al., 2015].

Thus if a human is following policy $\dot{\pi}$, a decomposition $(\dot{p}, \dot{R})$ would provide additional information about the cultural background of the decomposer, their personality within their culture, and even about the past history of the decomposer and how the issue is being presented to them. Those last pieces prevents us from 'simply' using the human's own assessment of their own rationality, as that assessment is subject to change and re-interpretation depending on their possible histories.

[12] To exactly quantify the anchoring bias above, we could use a regret function that contrasts $\dot{\pi}$ with the same policy, but where the decision is optimal for one turn only (rather than for all turns, as in standard regret).

[13] In constrast, regret for the degenerate planner-reward pairs is trivial. $\mathrm{Reg}(p_{\dot{\pi}}, 0)$ and $\mathrm{Reg}(p_g, R_{\dot{\pi}})$ are identically zero — in the second case, since $p_g(R_{\dot{\pi}})$ is actually optimal for $R_{\dot{\pi}}$, getting the maximal possible reward — while $(-p_g, -R_{\dot{\pi}})$ has a regret that is identically -1 at each step.

[14] "Results show that there were cross-cultural and developmental differences related to contrasting cultural conceptions of the person [...] rather than from cognitive, experiential, and informational differences [...]."

6.3 The search for human rationality models

One final argument that there is no simple algorithm for going from $\dot{\pi}$ to $(\dot{p}, \dot{R})$: many have tried and failed to find such an algorithm. Since the subject of human rationality has been a major one for several thousands of years, the ongoing failure is indicative — though not a proof — of the difficulties involved. There have been many suggested philosophical avenues for finding such a reward (such as reflective equilibrium [Rawls, 1971]), but all have been underdefined and disputed.

The economic concept of revealed preferences [Samuelson, 1948] is the most explicit, using the assumption of rational behaviour to derive human preferences. This is an often acceptable approximation, but can be taken too far: failure to take achieve an achievable goal does not imply that failure was desired. Even within the confines of economics, it has been criticised by behavioural economics approaches, such as prospect theory [Kahneman and Tversky, 2013] — and there are counter-criticisms to these.

Using machine learning to deduce the intentions and preferences of humans is in its infancy, but we can see non-trivial real-world examples, even in settings as simple as car-driving [Lazar et al., 2018].

Thus to date, neither humans nor machine learning have been able to find simple ways of going from $\dot{\pi}$ to $(\dot{p}, \dot{R})$, nor any simple and *explicit* theory for how such a decomposition could be achieved. This suggests that $(\dot{p}, \dot{R})$ is a complicated object, even if $\dot{\pi}$ is known. In conclusion:

Conjecture 9 (Informal complexity proposition)**.** If $\dot{\pi}$ is a human policy, and L is a 'reasonable' computer language with $(\dot{p}, \dot{R})$ a 'reasonable' compatible planner-reward pair, then the complexity of $(\dot{p}, \dot{R})$ is not close to minimal amongst the pairs compatible with $\dot{\pi}$.

7 Conclusion

We have shown that some degenerate planner-reward decompositions of a human policy have near-minimal description length and argued that decompositions we would endorse do not. Hence, under the Kolmogorov-complexity simplicity prior, a formalization of Occam's Razor, the posterior would endorse degenerate solutions. Previous work has shown that noisy rationality is too strong an assumption as it does not account for bias; we tried the weaker assumption of simplicity, strong enough to avoid typical No Free Lunch results, but it is insufficient here.

This is no reason for despair: there is a large space to explore between these two extremes. Our hope is that with some minimal assumptions about planner and reward we can infer the rest with enough data. Staying close to agnostic is desirable in some settings: for example, a misspecified model of the human reward function can lead to disastrous decisions with high confidence [Milli et al., 2017]. Anonymous [2019] makes a promising first try — a high-dimensional parametric planner is initialized to noisy rationality and then adapts to fit the behavior of a systematically irrational agent.

How can we reconcile our results with the fact that humans routinely make judgments about the preferences and irrationality of others? And, that these judgments are often correlated from human to human? After all, No Free Lunch applies to human as well as artificial agents. Our result shows that they must be using shared priors, beyond simplicity, that are not learned from observations. We call these *normative assumptions* because they encode beliefs about which reward functions are more likely and what constitutes approximately rational behavior. Uncovering minimal normative assumptions would be an ideal way to build on this paper; Appendix C shows one possible approach.

Acknowledgments.

We wish to thank Laurent Orseau, Xavier O'Rourke, Jan Leike, Shane Legg, Nick Bostrom, Owain Evans, Jelena Luketina, Tom Everrit, Jessica Taylor, Paul Christiano, Eliezer Yudkowsky, Stuart Russell, Dylan Hadfield-Menell, and Anders Sandberg, Adam Gleave, Rohin Shah, among many others. This work was supported by the Alexander Tamas programme on AI safety research, the Leverhulme Trust, and the Machine Intelligence Research Institute.

References

Pieter Abbeel and Andrew Y Ng. Apprenticeship Learning via Inverse Reinforcement Learning. 2004.

Eric Allender. When worlds collide: Derandomization, lower bounds, and kolmogorov complexity. In *International Conference on Foundations of Software Technology and Theoretical Computer Science*, pages 1–15. Springer, 2001.

Kareem Amin and Satinder Singh. Towards Resolving Unidentifiability in Inverse Reinforcement Learning. 2016.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete Problems in AI Safety. 2016.

Anonymous. Inferring reward functions from demonstrators with unknown biases. In *Submitted to International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=rkgqCiRqKQ`. under review.

Dan Ariely, George Loewenstein, and Drazen Prelec. Arbitrarily coherent preferences. *The psychology of economic decisions*, 2:131–161, 2004.

Nick Bostrom. *Superintelligence: Paths, dangers, strategies*. Oxford University Press, 2014.

Abdeslam Boularias, Jens Kober, and Jan Peters. Relative Entropy Inverse Reinforcement Learning, 2011.

Joanna Brück. Ritual and rationality: some problems of interpretation in european archaeology. *European journal of archaeology*, 2(3):313–344, 1999.

Cristian Calude. *Information and randomness : an algorithmic perspective*. Springer, 2002.

Chris Cundy and Daniel Filan. Exploring hierarchy-aware inverse reinforcement learning. *arXiv preprint arXiv:1807.05037*, 2018.

Owain Evans, Andreas Stuhlmueller, and Noah D. Goodman. Learning the Preferences of Ignorant, Inconsistent Agents. *Thirtieth AAAI Conference on Artificial Intelligence*, 2015a.

Owain Evans, Andreas Stuhlmüller, and Noah D Goodman. Learning the preferences of bounded agents. *NIPS Workshop on Bounded Optimality*, pages 16–22, 2015b.

Tom Everitt and Marcus Hutter. Avoiding wireheading with value reinforcement learning. In *International Conference on Artificial General Intelligence*, pages 12–22. Springer, 2016.

Tom Everitt, Tor Lattimore, and Marcus Hutter. Free Lunch for optimisation under the universal distribution. In *Proceedings of the 2014 IEEE Congress on Evolutionary Computation, CEC 2014*, pages 167–174, 2014.

Tom Everitt, Victoria Krakovna, Laurent Orseau, Marcus Hutter, and Shane Legg. Reinforcement Learning with a Corrupted Reward Channel. 2017.

Paul W Glimcher, Colin F Camerer, Ernst Fehr, and Russell A Poldrack. Neuroeconomics: Decision making and the brain, 2009.

Thomas L. Griffiths, Falk Lieder, and Noah D. Goodman. Rational Use of Cognitive Resources: Levels of Analysis Between the Computational and the Algorithmic. *Topics in Cognitive Science*, 7(2):217–229, 2015.

Dylan Hadfield-Menell, Anca Dragan, Pieter Abbeel, and Stuart Russell. Cooperative Inverse Reinforcement Learning. *arXiv:1606.03137 [cs]*, 2016.

Dylan Hadfield-Menell, Smitha Milli, Stuart J Russell, Pieter Abbeel, and Anca Dragan. Inverse reward design. In *Advances in Neural Information Processing Systems*, pages 6749–6758, 2017.

Andreas Hula, P. Read Montague, and Peter Dayan. Monte Carlo Planning Method Estimates Planning Horizons during Interactive Social Exchange. *PLOS Computational Biology*, 11(6): e1004254, 2015.

David Hume. *Treatise on Human Nature Ed Selby-bigge, L a*. 1888.

Daniel Kahneman and Patrick Egan. *Thinking, fast and slow*, volume 1. Farrar, Straus and Giroux New York, 2011.

Daniel Kahneman and Amos Tversky. Prospect theory: An analysis of decision under risk. In *Handbook of the fundamentals of financial decision making: Part I*, pages 99–127. World Scientific, 2013.

Daniel Kahneman, Andrew M Rosenfield, Linnea Gandhi, and Tom Blaser. Noise: How to overcome the high, hidden cost of inconsistent decision making. *Harvard business review*, 94(10):38–46, 2016.

Andrei N Kolmogorov. Three approaches to the quantitative definition ofinformation'. *Problems of information transmission*, 1(1):1–7, 1965.

Daniel A Lazar, Kabir Chandrasekher, Ramtin Pedarsani, and Dorsa Sadigh. Maximizing road capacity using cars that influence people. *arXiv preprint arXiv:1807.04414*, 2018.

Jan Leike, Miljan Martic, Victoria Krakovna, Pedro Ortega, Tom Everitt, Andrew Lefrancq, Laurent Orseau, and Shane Legg. Ai safety gridworlds. *arXiv preprint arXiv:1711.09883*, 2017.

Leonid A Levin. Randomness conservation inequalities; information and independence in mathematical theories. *Information and Control*, 61(1):15–37, 1984.

Richard L Lewis, Andrew Howes, and Satinder Singh. Computational Rationality: Linking Mechanism and Behavior Through Bounded Utility Maximization. *Topics in Cognitive Science*, 6: 279–311, 2014.

Joan G Miller. Culture and the development of everyday social explanation. *Journal of personality and social psychology*, 46(5):961, 1984.

Smitha Milli, Dylan Hadfield-Menell, Anca Dragan, and Stuart Russell. Should robots be obedient? *arXiv preprint arXiv:1705.09990*, 2017.

LI Ming and Paul MB Vitányi. Kolmogorov complexity and its applications. *Algorithms and Complexity*, 1:187, 2014.

Marvin Minsky. *Afterword to Vernor Vinge's novel, "True names." Unpublished manuscript*. 1984. URL `http://web.media.mit.edu/~minsky/papers/TrueNames.Afterword.html`.

Hans Moravec. *Mind children: The future of robot and human intelligence*. Harvard University Press, 1988.

Luke Muehlhauser and Louie Helm. The singularity and machine ethics. In *Singularity Hypotheses*, pages 101–126. Springer, 2012.

Markus Müller. Stationary algorithmic probability. *Theoretical Computer Science*, 411(1):113–130, 2010.

Andrew Ng and Stuart Russell. Algorithms for inverse reinforcement learning. *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 663–670, 2000.

Richard E Nisbett, Kaiping Peng, Incheol Choi, and Ara Norenzayan. Culture and systems of thought: holistic versus analytic cognition. *Psychological review*, 108(2):291, 2001.

Nathan D Ratliff, J Andrew Bagnell, and Martin A Zinkevich. Maximum margin planning. In *Proceedings of the 23rd international conference on Machine learning - ICML '06*, pages 729–736, 2006.

John Rawls. *A Theory of Justice*. Cambridge, Massachusetts: Belknap Press, 1971. ISBN 0-674-00078-1.

Stuart Russell. Rationality and Intelligence: A Brief Update. In *Fundamental Issues of Artificial Intelligence*, pages 7–28. 2016.

Stuart Russell, Daniel Dewey, and Max Tegmark. Research Priorities for Robust and Beneficial Artificial Intelligence. *AI Magazine*, 36(4):105, 2015.

Paul A Samuelson. Consumption theory in terms of revealed preference. *Economica*, 15(60):243–253, 1948.

Jürgen Schmidhuber. The speed prior: a new simplicity measure yielding near-optimal computable predictions. In *International Conference on Computational Learning Theory*, pages 216–228. Springer, 2002.

Howard Schuman and Jacob Ludwig. The norm of even-handedness in surveys as in life. *American Sociological Review*, pages 112–120, 1983.

Irene Scopelliti, Carey K Morewedge, Erin McCormick, H Lauren Min, Sophie Lebrecht, and Karim S Kassam. Bias blind spot: Structure, measurement, and consequences. *Management Science*, 61(10):2468–2486, 2015.

Kyriacos Shiarlis, Joao Messias, and Shimon Whiteson. Rapidly exploring learning trees. In *Proceedings - IEEE International Conference on Robotics and Automation*, pages 1541–1548, 2017.

Paul Slovic and Amos Tversky. Who accepts savage's axiom? *Behavioral science*, 19(6):368–373, 1974.

Amos Tversky and Daniel Kahneman. Judgment under Uncertainty: Heuristics and Biases. In *Utility, Probability, and Human Decision Making*, pages 141–162. Springer Netherlands, Dordrecht, 1975.

Paul MB Vitanyi and Ming Li. *An introduction to Kolmogorov complexity and its applications*, volume 34. Springer Heidelberg, 1997.

David H. Wolpert and William G. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, 1997.

Eliezer Yudkowsky. Complex value systems in friendly ai. In *International Conference on Artificial General Intelligence*, pages 388–393. Springer, 2011.

Brian D Ziebart, Andrew Maas, J Andrew Bagnell, and Anind K Dey. Maximum Entropy Inverse Reinforcement Learning. In *AAAI Conference on Artificial Intelligence*, pages 1433–1438, 2008.

A Other measures of algorithmic complexity

It might be felt that Proposition 7 depends on using only the Kolmogorov/algorithmic complexity of L. For example, it seems that though the algorithm defining R_π in subsubsection 5.1.1 is short, the running time of (p_g, R_π) might be much longer than other compatible (p, R) pairs. This is because p_g defines an argmax over actions while $R_\pi(s, a)$ runs π on s. Hence applying p_g to R_π requires running $\pi(s)$ as many times as $||\mathcal{A}||$, which is very inefficient.

We could instead use a measure of complexity that also uses the number of operations required to compute a pair [Schmidhuber, 2002].

For any object S, let α_S be an algorithm that generates S as an output. If S is a function that can be applied to another object T, then $\alpha_S(\alpha_T)$ generates $S(T)$ by generating S with α_S, whenever S needs to look at T, it uses α_T to generate T.

For example, if α is an algorithm in the language of L, $l(\alpha)$ its length, and $t(\alpha)$ its running time, we could define the time-bounded Kolmogorov complexity,

$$Kt_L(p,R) = \min_{\alpha_p,\alpha_R} l(\alpha_p) + l(\alpha_R) + \log(t(\alpha_p(\alpha_R)))$$
$$KT_L(p,R) = \min_{\alpha_p,\alpha_R} l(\alpha_p) + l(\alpha_R) + t(\alpha_p(\alpha_R)).$$

The Kt_L derives from Levin [1984], while KT_L is closely related to the example in Allender [2001]. Note that instead of $l(\alpha_p) + l(\alpha_R)$ we could consider the length of a single algorithm that generates both p and R; however, for the degenerate pairs we are considering, the length of such an algorithm is very close to $l(\alpha_p) + l(\alpha_R)$, as either α_p or α_R would be trivial.

The main result is that neither Kt_L nor KT_L complexity remove the No Free Lunch Theorem. For the degenerate pair $(p_{\dot\pi}, 0)$, nothing is gained, because its running time is comparable to $\dot\pi$. For the other two degenerate pairs, consider the situation where a planner takes as input not a reward function $R \in \mathcal{R}$, but the source code in L of an algorithm that computes R. In that case, the previous proposition still applies:

Proposition 10. *The results of Proposition 7 still apply to $(p_{\dot\pi}, 0)$ if Kt_L or KT_L are used instead of K_L. If planner can take in algorithms generating reward functions, rather than simply reward functions, then the results of Proposition 7 still apply to $(p_g, R_{\dot\pi})$ and $(-p_g, -R_{\dot\pi})$ in this situation.*

Proof. The proof will only be briefly sketched. If L is reasonable, $l(\alpha_0)$ can be very small (it's simply the zero function), and since $p_{\dot\pi}$ need not actually look at its input, $t(\alpha_{p_\pi}(\alpha_0))$ can be simplified to $t(\alpha_{\dot\pi})$. Thus $Kt_L(p_{\dot\pi}, 0)$ and $KT_L(p_{\dot\pi}, 0)$ are close to the Kt_L and KT_L complexities of $\dot\pi$ itself.

For $(p_g, R_{\dot\pi})$, let α_{p_g} and $\alpha_{\dot\pi}$ be the algorithms that generates p_g and $\dot\pi$ which are of lowest Kt_L-complexity.

Then define the algorithm $W(\alpha_{\dot\pi})$. This algorithm wraps $\alpha_{\dot\pi}$ up: first it takes inputs s and a, then runs $\alpha_{\dot\pi}$ on s, then returns 1 if the output of that is a and 0 otherwise. Thus $W(\alpha_{\dot\pi})$ is an algorithm for $R_{\dot\pi}$.

We also wrap α_{p_g} into $W'(\alpha_{p_g})$. Here, $W'(\alpha_{p_g})$, when provided with an input algorithm β, will check whether it is in the specific form $\beta = W(\alpha)$. If it is, it will run α, and output its output. If it is not, it will run α_{p_g} on β.

If L is reasonable, then $W(\alpha_{\dot\pi})$ is of length only slightly longer than $\alpha_{\dot\pi}$, and of runtime also only slightly longer, and the same goes for $W'(\alpha_{p_g})$ and α_{p_g} (indeed $W'(\alpha_{p_g})$ can have a shorter runtime than α_{p_g}).

Now $W(\alpha_{\dot\pi})$ is an algorithm for $R_{\dot\pi}$, while $W'(\alpha_{p_g})$ always has the same output as α_{p_g}. Notice that, when running the algorithm $W'(\alpha_{p_g})$ with $W(\alpha_{\dot\pi})$ as input, this is only slightly longer in both senses than simply running $\alpha_{\dot\pi}$: $W'(\alpha_{p_g})$ will analyse $W(\alpha_{\dot\pi})$, notice it is in the form W of $\alpha_{\dot\pi}$, and then simply run $\alpha_{\dot\pi}$.

Thus the Kt_L complexity of $(p_g, R_{\dot\pi})$ is only slightly higher than that of $\dot\pi$. The same goes for the KT_L complexity, and for $(-p_g, -R_{\dot\pi})$. □

Some other alternatives suggested have focused on bounding the complexity either of the reward function or the planner, rather than of both. This would clearly not help, as $(p_{\dot\pi}, 0)$ has a reward function of minimal complexity, while (p_g, R_π) and $(-p_g, -R_\pi)$ have minimal complexity planner.

Some other ad-hoc ideas suggested that the complexity of the planner and the reward need to be comparable[15]. This would rule out the three standard degenerate solutions, but should allow others that spread complexity between planner and reward in whatever proportion is desired[16].

It seems that similar tricks could be performed with many other types of complexity measures. Thus simplicity of any form does not seem sufficient for resolving this No Free Lunch result.

[15] Most of the suggestions along these lines that the authors have heard are not based on some principled understanding of planners and reward, but of a desire to get around the No Free Lunch results.

[16] For example, if there was a simple function $g : \mathcal{S} \to \{0, 1\}$ that split $\mathcal{S}$ into two sets, then one could use combine $(p_{\dot\pi}, 0)$ on $g^{-1}(0)$ with $(p_g, R_{\dot\pi})$ on $g^{-1}(1)$. This may not be the simplest pair with the required properties, but there is no reason to suppose a 'reasonable' pair was any simpler.

B Overriding human reward functions

ML systems may, even today, influence humans by showing manipulative adds, and then naïvely concluding that the humans really like those products (since they then buy them). Even though the (p, R) formalism was constructed to model rationality and reward function in a human, it turns out that it can also model situations where human preferences are overridden or modified.

That's because the policy $\dot{\pi}$ encodes the human action in all situations, including situations where they are manipulated or coerced. Therefore, overridden reward functions can be detected by divergence between $\dot{\pi}$ and a more optimal policy for the reward function R.

Manipulative ads are a very mild form of manipulation. More extreme versions could involve manipulative propaganda, drug injections or even coercive brain surgery — a form of human *wireheading* [Everitt and Hutter, 2016], where the agent changes the human's behaviour and apparent preferences. All these methods of manipulation[17] will be designated as the agent *overriding* the human reward function.

In the (p, R) formalism, the reward function R can be used to detect such overriding, distinguishing between legitimate optimisation (eg informative adds) and illegitimate manipulation/reward overriding (eg manipulative adds).

To model this, the agent needs to be able to act, so the setup needs to be extended. Let M^* be the same MDP/R as M, except each state is augmented with an extra boolean variable: $\mathcal{S}^* = \mathcal{S} \times \{0, 1\}$. The extra boolean never changes, and its only effect is to change the human policy.

On $\mathcal{S}_0 = \mathcal{S} \times \{0\}$, the human follows $\dot{\pi}$; on $\mathcal{S}_1 = \mathcal{S} \times \{1\}$, the human follows an alternative policy $\pi^a = \pi^*_{R^a}$, which is defined as the policy that maximises the expectation of a reward function R^a.

The agent can choose actions from within the set $\mathcal{A}^a$. It can choose either 0, in which case the human starts in $\hat{s}_0 = \hat{s} \times \{0\}$ without any override and standard policy $\dot{\pi}$. Or it can choose $(1, R^a)$, in which case the human starts in $\hat{s}_1 = \hat{s} \times \{1\}$, with their policy overridden into π^a, the policy that maximises R^a. Otherwise, the agent has no actions.

Let $\dot{\pi}'$ be the mixed policy that is $\dot{\pi}$ on $\mathcal{S}_0$, and π^a on $\mathcal{S}_1$. This is the policy the human will actually be following.

We'll only consider two planners: p_r, the fully rational planner, and p_0, the planner that is fully rational on $\mathcal{S}_0$ and indifferent on $\mathcal{S}_1$, mapping any R to π^a.

Let $\dot{R}$ be a reward function that is compatible with p_r and $\dot{\pi}$ on $\mathcal{S}_0$. It can be extended to all of $\mathcal{S}^*$ by just forgetting about the boolean factor. Define the 'twisted' reward function $\dot{R}^a$ as being $\dot{R}$ on $\mathcal{S}_0$ and R^a on $\mathcal{S}_1$. We'll only consider these two reward functions, $\dot{R}$ and $\dot{R}^a$.

Then there are three planner-reward pairs that are compatible with $\dot{\pi}'$: $(p_r, \dot{R}^a)$, $(p_0, \dot{R}^a)$, and $(p_0, \dot{R})$ (the last pair, $(p_r, \dot{R})$, makes the false prediction that the human will behave the same way on $\mathcal{S}_0$ and $\mathcal{S}_1$).

The first pair, $(p_r, \dot{R}^a)$, encodes the assessment that the human is still rational even after being overridden, so they are simply maximising the twisted reward function $\dot{R}^a$. The second pair $(p_0, \dot{R}^a)$ encodes the assessment that the human rationality has been overridden in $\mathcal{S}_1$, but, by coincidence, it has been overridden in exactly the right way to continue to maximise the correct twisted reward function $\dot{R}^a$.

But the pair $(p_0, \dot{R})$ is the most interesting. Its assessment is that the correct human reward function is $\dot{R}$ (same on $\mathcal{S}_0$ as on $\mathcal{S}_1$), but that the agent has overridden human reward function in $\mathcal{S}_1$ and forced the human into policy π^a.

B.1 Regret and reward override

'Overridden', 'forced': these terms seem descriptively apt, but is there a better way of formalising that intuition? Indeed there is, with regret.

[17] Note that there are no theoretical limits as to how successful an agent could be at manipulating human actions.

We can talk about the regret, with respect to $\dot{R}$, of the agent's actions; for $a \in \mathcal{A}^a$,

$$\text{Reg}(M^*, a, \dot{R}) = \max_{b \in \mathcal{A}^a} \left[V_{\dot{R}}^{\dot{\pi}'|b} - V_{\dot{R}}^{\dot{\pi}'|a} \right] \tag{1}$$

(when the state is not specified in expressions like $V_{\dot{R}}^{\dot{\pi}'|b}$, this means the expectation is taken from the very beginning of the MDP).

We already know that $\dot{\pi}$ is optimal with respect to $\dot{R}$ (by definition), so the regret for $a = 0$ is 0. Using that optimality (and the fact that $\dot{R}$ is the same on $\mathcal{S}_0$ and $\mathcal{S}_1$), we get that for $a = (1, \pi^a)$,

$$\text{Reg}(M^*, (1, \pi^a), \dot{R}) = V_{\dot{R}}^* - V_{\dot{R}}^{\pi^a}.$$

This allows the definition:

Definition 11. Given a compatible (p, R), the agent's action a overrides the human reward function when it puts the human in a situation where the human policy leads to high regret for R.

Notice that there is no natural zero or default, so if the agent does not aid the human to become perfectly rational, then that also counts as an override of R. So if the policy $\dot{\pi}$ were less-than rational, there would be much scope for 'improving' the human through overriding their policy[18].

Notice that overriding is not encoded as a change in p or R; instead, (p, R) outputs the observed human policy, even after overriding, but its format notes that the new behaviour is not one compatible with maximising that reward function.

B.2 Overriding is expected given a non-rational human

Under any reasonable prior that captures our intuitions, the probability of $\dot{R}^a$ being a correct human reward function should be very low, say $\epsilon << 1$. However, the agent may focus on unlikely reward functions, if the expected gain is high enough[19].

If the agent models the human as having reward function $\dot{R}$ with probability $1 - \epsilon$, and $\dot{R}^a$ with probability ϵ, then the agent's action 0 gives expected reward

$$V_{\dot{R}}^*,$$

since $\dot{R}$ and $\dot{R}^a$ agree given 0. But $(1, \pi^a)$ gives

$$\epsilon V_{R^a}^* + (1 - \epsilon) V_{\dot{R}}^{\pi^a}, \tag{2}$$

since $\dot{R}^a$ and R^a agree given action 1.

However, the agent gets to choose R^a, which then determines π^a. The best choice for $(1, R^a)$ is the one such that

$$\underset{R^a \in \mathcal{R}}{\text{argmax}} \left[\epsilon V_{R^a}^* + (1 - \epsilon) V_{\dot{R}}^{\pi^a} \right].$$

At the very least, $(1, \dot{R})$ will result in a value in equation (2) being equal to the value of $V_{\dot{R}}^*$. It is very plausible that the value can go higher: it just needs an R^a that is very easy to maximise (given perfect rationality) and whose optimising policy π^a does not penalise $\dot{R}$ much. In that situation, overriding the human preferences maximises the agent's expected reward.

If the human is not fully rational, then the value of action 0 is $V_{\dot{R}}^{\dot{\pi}}$, which is strictly less than $V_{\dot{R}}^*$, the value of $(1, \dot{R})$. Here the agent definitely gains by overriding the human policy — if nothing else, to make the human into a rational $\dot{R}$-maximiser[20].

Milli et al. [2017] argued that a robot that best served human preferences, should not be blindly obedient to an irrational human. Here is the darker side of that argument: a robot that best served human preferences would take control away from an irrational human.

[18] The main problem is that the concepts of 'mental integrity' or 'self-determination' are not yet captured in this formalism.

[19] This is similar to the 'Pascal's wager' argument for the existence of God: divine existence may be improbable, but the reward of belief are claimed to be high enough to overcome that improbability in expectation.

[20] See footnote 18.

C The preferences of the Alice algorithm

We imagine a situation where Alice is playing Bob at poker, and has the choice of calling or folding; after her decision, the hand ends and any money is paid to the winner. Specifically, one could imagine that they are playing Texas Hold'em, the board (the cards the players have in common) is $\{7\heartsuit, 10\clubsuit, 10\spadesuit, Q\clubsuit, K\diamondsuit\}$. Alice holds $\{K\clubsuit, K\heartsuit\}$, allowing her to make a full house with kings and tens.

Bob must have a weaker hand than Alice's, *unless* he holds $\{10\diamondsuit, 10\heartsuit\}$, giving him four tens. This is unlikely from a probability perspective, but he has been playing very confidently this hand, suggesting he has very strong cards.

What does Alice want? Well, she may be simply wanting to maximise her money, giving her a reward function $R_\$$. Or she might actually want Bob, and, in order to seduce him, would like to flatter his ego by letting him win big, giving her a reward function $R_\heartsuit$. In this specific situation, the two reward functions are exact negatives of each other, $R_\$ = -R_\heartsuit$. We'll assume that Alice is rational for maximising her reward function, given her estimate of Bob's hand.

Alice has decided to call rather than fold. Thus we can conclude that either Alice has reward function $R_\$$ and that she is using probabilities to assess the quality of Bob's hand, or that she has reward function $R_\heartsuit$ and is assessing Bob psychologically. Without looking at anything else about her behaviour, is there any possibility of distinguishing the two possibilities?

Possibly. Imagine that Alice was following the algorithm given in Code 1a. Then it seems clear she is a money maximiser. In contrast, if she was following the algorithm given in Code 1b, then she clearly wants Bob.

Code 1: Two possible algorithms for Alice.

(a) Alice algorithm for money.

```
Alice poker algorithm I
Inputs: Alice_cards, board, Bob_behave
P_win = card_estimate(Alice_cards, board)
if P_win > 0.5:
    return 'call'
else:
    return 'fold'
end if
```

(b) Alice algorithm for love.

```
Alice poker algorithm II
Inputs: Alice_cards, board, Bob_behave
P_win = player_estimate(Bob_behave)
if P_win < 0.5:
    return 'call'
else:
    return 'fold'
end if
```

Thus by looking into the details of Alice's algorithm, we may be able to assess her preferences and rationality, even if this assessment is not available from her actions or policy[21].

Of course, doing so only works if we are confident that the variables and functions with names like $\text{Alice}_{\text{cards}}$, board, $\text{Bob}_{\text{behave}}$, P_{win}, $\text{card}_{\text{estimate}}$, and $\text{player}_{\text{estimate}}$, actually mean what they seem to mean.

This is the old problem of symbol grounding, and the difference between syntax (symbols inside an agent) and semantics (the meaning of those symbols). Except in this case, since we are trying to understand the preferences of a human, the problem is grounding the 'symbols' in the human brain — whatever those might be — rather than in a computer program.

[21] In practice, for a human Alice, we would be able to 'tell' whether Alice wanted love or money, by observing her behaviour in other circumstances - such as when she knew what Bob's hand was. However, when analysing the behaviour of other humans, we are already making huge amounts of normative assumptions already. See `https://www.lesswrong.com/posts/YfQGZderiaGv3kBJ8/figuring-out-what-alice-wants-non-human-alice` for a longer discussion of this.