Inverse Reinforcement Learning (IRL) in Presence of Risk and Uncertainty Related Cognitive Biases

To what extent can IRL learn rewards from expert demonstrations with loss and risk aversion?

Meric Ikiz[1]

Supervisor(s): Luciano Cavalcante Siebert[1]**, Angelo Caregnato Neto**[1]

[1]EEMCS, Delft University of Technology, The Netherlands

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 25, 2023

Name of the student: Meric Ikiz
Final project course: CSE3000 Research Project
Thesis committee: Luciano Cavalcante Siebert, Angelo Caregnato Neto, Jana Weber

Abstract

A key issue in Reinforcement Learning (RL) research is the difficulty of defining rewards. Inverse Reinforcement Learning (IRL) is a technique that addresses this challenge by learning the rewards from expert demonstrations. In a realistic setting, expert demonstrations are collected from humans, and it is important to acknowledge that these demonstrations can deviate from rationality due to systematic biases known as cognitive biases. One group of cognitive biases, known as risk-sensitive cognitive biases, pertains to individuals' attitudes and behaviors towards risk and uncertainty. This paper investigates the extent to which IRL can learn from demonstrations that contain risk-sensitive cognitive biases such as loss aversion and risk aversion. Modelling biases using concepts from Prospect Theory and System 1 and 2 model and using Maximum Entropy IRL algorithm, this paper concludes that IRL can recreate similar solutions to experts but inferring the underlying motivations and the interactions between them is an intricate problem that requires novel approaches.

1 Introduction

An important issue in Reinforcement Learning (RL) research is that for many tasks it is hard to devise a concrete reward function that accurately describes the goal of the agent. Inverse Reinforcement Learning (IRL) is a method developed to circumvent this issue. In IRL the agent is given a set of expert demonstrations from which it is expected to recover the reward function and then find an optimal policy according to the inferred reward function [1].

This way of formulating the problem can improve the performance of machine learning algorithms on tasks that are hard to model. One example is driving. Even though creating demonstrations for driving is easy, describing the exact goals and the trade offs between these goals would be challenging. A multitude of examples can be found when thinking about tasks that are conventionally easy and intuitive for humans.

The process of inferring a reward function from human demonstrations presents many new and different challenges. One such challenge is that human behaviour is not completely rational and can contain noisy, even contradicting behaviours. Even though every instance of human behaviour will deviate in a unique way from what an algorithm might consider "optimal", research shows that there are some patterns that humans are susceptible towards [4] [13]. These patterns are called cognitive biases. Even though cognitive biases help humans by acting as "shortcuts in thinking" and decrease the computational resources needed to make a decision, their presence in expert demonstrations should be accounted for in order to construct a realistic reward function.

This paper focuses on the cognitive bias called loss aversion. Loss aversion reflects the skewed risk and uncertainty perception of humans, specifically the tendency to overestimate losses and avoid risk even if it is at the expense of the reward. This paper aims to answer the following question: "To what extent can IRL learn rewards from demonstrations that contain loss aversion bias?".

This study adds value to the field of IRL by studying an essential component in human decision making: perception of risk and uncertainty. By bridging concepts from Economics, Psychology and Cognitive Science into the realm of Computer Science, we integrate valuable research from other fields. Through assessing the applicability of such models, we contribute to building more explainable models.

Section 2 offers more insights to this research field while section 3 explains how this paper simulates cognitive biases by making use of Prospect Theory [5] [16] and System 1 and 2 Cognitive Model [4]. Section 4 conducts experiments using biased expert agent demonstrations to infer rewards through Maximum Entropy IRL (MEIRL) algorithm [17] and shares the results. Finally the sections 5 and 7 conducts a more in depth discussion about this experiment, drawing conclusions and suggesting future directions.

2 Background

The field of IRL is a relatively novel technique popularized in the year 2000 by paper [8]. This paper highlights the problem of reward definition in RL and proposes IRL as a solution, deriving the algorithm from the very basic representation of MDPs and Bellman Equations, which are the building blocks of most RL algorithms. This work considered the expert demonstrations as optimal, which might not always be the case. Research shows that demonstrated behaviour and underlying intentions can often be very distinct [14] and this has traditionally been left out of technical fields.

In engineering, human models have traditionally assumed that humans are rational actors who act according to expected utility. Previous research in combining cognitive models and Inverse Reinforcement Learning (IRL) has often followed a probability-based approach and considered incorporating a component that represents the degree of rationality of the agent [2]. However, interpreting observed behavior solely as "deviating from rationality" fails to capture the full complexity of the situation. While accounting for biases that may hinder certain goals is important, it does not fully explain behavior observed in demonstrations.

The paper [9] describes models that represent the interplay between different goals in single choice and continuous contexts. This is an approach inspired by Cognitive Science, and specifically the mechanism of System 1 and 2 described in [4]. System 2 represents the rational choices while System 1 is more affected by the "shortcuts in thinking". The paper is very through with the temporal interaction of these two systems as it acknowledges that a decision made at a certain time affects the nature of the choices in later stages.

Modelling uncertainty, a prevalent model is Prospect Theory [5] [16], used in Cognitive Science and Economics to model human behaviour in reaction to uncertainty and risk taking. Prospect Theory presents mathematical models that captures cognitive biases that describe an agent that treats losses and gains asymmetrically, overestimates the probability of low probability events and underestimates the proba-

bility of high probability events. The model also shows that additional losses or additional gains after a point are less relevant in how the agent perceives the reward. To illustrate one can think of the example that the perceived value difference between \$10 and \$20 is higher than for \$100 and \$110.

The IRL method that will be used on this paper is Maximum Entropy IRL (MEIRL). The main challenge in inferring reward function is that for a set of given demonstration there are infinitely many reward functions that can lead to similar behaviour. The popular method of MEIRL chooses the solution according to the principle of maximum entropy, that the solution that has the maximum entropy is the solution that fits the observation with the least amount of introduced bias [18]. The reward prediction is made using a forward and a backward pass in the algorithm observing the actions given state and state features and a forward pass computing state visitation frequencies. These passes are executed interchangeably until convergence (or close to convergence).

3 Methodology

The process followed in experiments in this paper can be examined in three parts. First, the expert demonstrations are generated that reflect the risk sensitive cognitive biases and multiple reward functions acting jointly. Second, MEIRL algorithm is used to infer expert goals and preferences. Third, the policy generated by the learning agent is compared to the expert policy in different environments and differences are observed. This section explains the first two steps. After presenting the final algorithm used to generate the model of preferences of the expert, the MEIRL algorithm is explained. The evaluation of the learning IRL agent is briefly discussed in this section. However more detailed discussion about the experiments and evaluation methods can be found in later sections 4 and 5.

3.1 Use of System 1 and 2 Cognitive Model

Modelling the expert agent, the System 1 and 2 model, described in [4] [9] is used to capture different preferences present within a person. The System 1 and 2 model is meant to represent two forms of human thought processes making that work jointly to determine the final decision. System 1 is more intuitive, heuristic-based, and uses shortcuts; it makes faster decisions with less consideration to long-term impacts. This also makes System 1 more open to biases. Meanwhile, System 2 is closer to the conventional rational-actor human model, which is used to model humans in engineering as long-term oriented decision makers. System 2 is therefore more detail based than System 1, and makes decisions significantly slower. The decision made by a human actor is always an interplay between irrational biases and intuition versus long-term thinking and rationality. Therefore, the interplay between System 1 and System 2 together form a more realistic representation of human decision making than a singe reward would.

Before we introduce how the two systems work together, we should look at how conventional value iteration works. In normal value iteration, there is a value table that is iterated through and updated by considering possible states and actions, along with both immediate and long-term expected rewards. This continues until the values in the table no longer change significantly (this is referred to as "convergence"). The long term rewards and discounted by a discount factor, while immediate rewards are interpreted at face value. This means that future rewards are valued less than short-term rewards. At a given point, normal value iteration updates the value table making use of the below rule where s represents the current state iterated in the value iteration algorithm and $Q(s,a)$ is calculated for all possible actions a that can be taken from state s.

$$V(s) = \max_a Q(s,a), \quad \text{where } Q(s,a) = r(s,a) + \gamma \cdot V(s_{\text{next}}) \tag{1}$$

The term $V(s_{\text{next}})$ is the expected cumulative reward starting from the next state that the selected action leads to and is discounted by the factor γ.

The paper [9] adapts this algorithm to model decision making when the agent is under the influence of two reward functions, with each having a their own discount factor. The models simulates the interplay between the systems by updating the individual value tables interchangeably. This means System 1 is updated based on the latest version of System 2 value table, and vice versa. The idea behind this is to have both Systems acknowledge each others preferences, and thereby make a decision that is ultimately a result of the codependency between these systems. Furthermore, this combined value iteration introduces cognitive control cost, which represents the cost of deviating from what would be the optimal course of action according to System 1. For a single decision the resulting reward can be computed as:

$$r = r_2(a) + \Psi(r_1(a^*) - r_1(a)) \tag{2}$$

where $r_1(a)$ and $r_2(a)$ are the rewards obtained from System 1 and 2 with the chosen action and $r_1(a^*)$ is the maximum obtainable reward from System 1, $a*$ representing the optimal action solely by System 1's perspective. The difference between the System 1 rewards from optimal and chosen actions is put through Ψ function and represents the cognitive control cost. It can be any function in theory, this paper assumes a linear term, referring to the resulting term as a cognitive control constant.

When the two systems are aware of each other and the actions are considered with their impacts on future rewards, the update rule becomes:

$$V_{\text{combined}}(s,\pi) = V_2(s,\pi) - \psi(V_1(s,\pi_1^*) - V_1(s,\pi)) \tag{3}$$

where π represents the combined policy and π_1^* represents System 1's optimal policy.

In the model implemented in this paper, only System 1 evaluation is filtered through Prospect Theory (explained in the following section). This is, in truth, not a realistic assumption; however, for the purposes of this paper, the biased attitudes towards risk and uncertainty presented by real humans is more closely represented by System 1. The combination of two systems with Prospect Theory is elaborated in section 3.3, offering the final algorithm used for expert's value table.

3.2 Introducing Uncertainty: Use of Prospect Theory

Part of the model used in this paper is inspired by the ideas and equations on Prospect Theory [5] [16] and used in the simplest form that still captures observations on human behaviour and biases like risk and loss aversion.

Subjective Assessment of Rewards

Reward can be either positive (gain) or negative (loss). Research shows that people perceive rewards relative to a reference value and react more strongly to losses than gains [15]. This gives rise to an S shaped piece-wise function that classify a reward as gain or loss relative to a reference/baseline point indicated by b in the equation.

$$r_{\text{subj}}(x) = \begin{cases} (x-b)^{\alpha} & \text{if } x > b, \\ -\kappa(b-x)^{\beta} & \text{if } x < b. \end{cases} \tag{4}$$

The variable x is the outcome for which the subjective reward is calculated. Simply one can call this objective reward in the context of this paper. The baseline value b benefits from further explanation. For example in a finance investment setting, b can be the initial amount of money one started with. In another context, if the agent is on a health journey b can represent their current weight, or their targets or aspirations like the goal weight or a certain calorie intake they planned for that day. In a lot of cases, the reference value can just be 0 and the gains and losses can be perceived in their absolute reward magnitude. Variables α and β determine the curvature of the function for gains and losses respectively. This represents the diminishing effect of further gains and losses. For example if the agent collected 10 units of rewards, collecting a 1 unit rewards is less attractive compared to an agent who did not collect any reward at that point. Finally κ is a positive constant that represents the degree of risk sensitivity to losses. This variable demonstrates the loss aversion bias where a certain amount of loss is perceived more strongly than an equivalent amount of gain. For example 1 unit of punishment (negative reward) is perceived more serious than 1 unit of gain. This can lead to the agent missing out on future rewards because of the initial negative reward.

Subjective Assessment of Uncertainty

According to Prospect Theory, human behaviour does not only exhibit loss aversion but also risk aversion and ambiguity aversion [16]. This means that the probabilities are not taken as is but filtered through a subjective lens of risk assessment. The subjective probability of events can be modeled as a weighting applied on the true probabilities. The weighting function is chosen to be:

$$w(p) = \frac{p^{\eta}}{p^{\eta} + (1-p)^{\eta}} \tag{5}$$

where p, the true probability of an event occurring is mapped to a subjective assessment of probability called "decision weight" p_{subj} [5]. This value is not a probability as it does not satisfy the axioms. For example the decision weights for the outcomes of a particular event does not necessarily sum to 1. However they are normalized before usage due to the nature of the used algorithms.

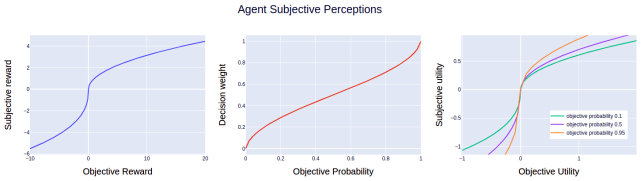

Figure 1: Perception of utility by the agent

One should keep in mind that this is a simple formulation of decision weights and subjective assessment of rewards as one key factor is left out of this paper's scope. This is the subjective belief of the agent about the likelihood of a particular event at any time and the personal values they associate with an event representing their personal reward preference. We propose this as a future improvement in section 7.

3.3 Integration of System 1 and 2 Model and Prospect Theory

Incorporating the cognitive model theories described above to make a single decision making algorithm, we apply the subjective assessment of rewards and uncertainty to System 1 reward values and probabilities. These are represented by $r_{\text{subj}}(x)$ and $w(p)$ and described in equations 4 and 5 respectively. This subjective assessment by System 1 is integrated with the modified value iteration described in equation 3. The final algorithm is below.

Algorithm 1 Final Algorithm Used by the Expert

Update r_1 rewards using r_{subj}
Update System 1 probabilities to decision weights using $w(p)$
Compute V_1^* solely adhering to System 1 preferences
Initialize V_1 and V_2 arbitrarily
Flip $\leftarrow$ True
while V_1 not converged$\|V_2$ not converged **do**
  for $s \in S$ **do**
    for $a \in A$ **do**
      $Q_1(s,a) \leftarrow w(p) * r_1(s) + \gamma_1 V_1(\tau(s,a))$
      $Q_2(s,a) \leftarrow r_2(s) + \gamma_2 V_2(\tau(s,a))$
    end for
    Let $a_{\max} = \arg\max_a [\psi Q_1(s,a) + Q_2(s,a)]$
    if Flip **then**
      Set $V_1(s) \leftarrow Q_1(s, a_{\max})$
    else
      Set $V_2(s) \leftarrow Q_2(s, a_{\max})$
    end if
    Flip $\leftarrow$ **not** Flip
  end for
end while
$V_{\text{final}} \leftarrow V_2 - \psi(V_1^* - V_1)$
return V_{final}

3.4 Learning Policy from Expert Demonstrations using Maximum Entropy IRL Algorithm

IRL aims to find the reward function that maximizes the likelihood of the observed expert behaviour. As described in 2

this is an ill-posed problem and solved by the use of Principle of Maximum Entropy. For more information, reader can consult the papers [18], [17]. In this paper there is a relevant part in section 2 and a visual representation in section 4.

The baseline implementation of the MEIRL algorithm is taken from the Github Repository [6]. To fit the purpose of this paper, the MEIRL algorithm has been extended by adding an environment that supports several reward functions rather than one (System 1 and 2), and the agent having its subjective perception of the environment (Prospect Theory). Several parts of the code are modified to work with the extra functionality introduced. One example to this is the new value iteration algorithm 1.

Expert and Learning Agent Comparison Methods

In order to evaluate how well the agent learns from demonstrated behaviour, this paper makes use of qualitative and quantitative assessment methods. The methods are described briefly in below list. More information is provided in later sections 4 and 5.

- **Qualitative assessment of trajectory preferences made by expert and learning agent:** Different scenarios are experimented with and commented on to highlight the preferences of agents in presence and absence of cognitive model and various critical decision points. In the context of this paper, this evaluation is needed since the topic of cognitive biases is inherently complex and contains elements of subjectivity that can best be described by human commentary. Solely comparing trajectories by similarity in the states they go through would result in inadequate analysis.
- **Comparison of rewards achieved by the expert and the comparison:** One of the quantitative metrics used is comparing the absolute magnitude of rewards collected through chosen trajectories. Rewards from System 1, 2 as well as the combination is compared using the objective and subjective utility of the rewards.
- **Expert and Learning agent policy comparison:** The policy matrices created by the expert and agent are compared using average cosine similarity between matrices.

4 Experiments and Results

The implementation for experimental setups in this paper can be found in GitHub [3].

4.1 Grid World Environment Description

We conducted experiments on cognitive model and IRL algorithm through the use of grid worlds where the possible actions are right, left, up and down and states are the position of the agent. In this grid world the actions always lead to their intended next states. This means if the agent's state is (1, 2) in coordinates the action is chosen as up, the agent's next state will be (1, 3). However the agent might not be able to reach every state and not all actions might always be available. This is represented by roads in the grid world (optional) that the agent can move in. There are two different rewards defined, for system 1 and system (referred to as $R1$ and $R2$ respectively). Although the end reward is simply the sum of these rewards, the expert chooses rewards using the algorithm described in section 3, where both systems are present at each decision making point with their separate goals. The rewards of System 1 are probabilistic, meaning the reward is received with a probability while $R2$ rewards are deterministic (received with a probability of 1.0). The assumption here is that the agent has perfect information of the world and without the input of System 1, would behave perfectly rationally. This is not realistic and merely a design choice intended to simplify experimental setup and observe the effect of System 1 in isolation.

4.2 Experiment 0: Proof of Concept Small Grid World

First, as a proof of concept, we demonstrate the steps discussed in the section 3 on a simple problem. The setup for this problem is shown on figure 2. The figure flow shows the experimental process, starting from the input rewards (indicated in green) and ends with the reward recovered by using MEIRL.

The small grid world consists of 25 states and roads do not exist, meaning the agent can reach every state. System 1 has a big reward for the top and a small reward for bottom right corner, while System 2 has a big reward on bottom and a very small reward on top right corner. Additionally System 1 has a set of negative rewards blocking the way to the right bottom corner, that is received with a probability. The agent starts from the middle of the left side and can terminate episode in top or bottom right corner. This design choice simplifies the analysis as we can directly see which system had a larger impact on the final course. The negative values in $R1$ are included to demonstrate expert's behaviour in situations that involve risk. While the larger rewards are solely intended to reflect the preferences of the system, the smaller rewards have a practical purpose too. They encourage the separate reward systems to have some motivation to travel.

In the experiment described in detail in figure 2, the main objective is to show the effect of cognitive control cost in decision making. Considering the agent starts from the middle of the left side, we can see that he opts for either of two terminal states ($T1$ for what is favorable for System 2 and $T2$ for otherwise) in about half of the trials, with $T1$ having a slightly higher chance. Even though the System 2 agent wants to go there but is hesitant because of System 1, when the agent ends up in $T2$, he arrives there very suboptimally. Often he passes through 3 punishment states when it is possible to pass through just 1. This is because of the cognitive control constant, causing the agent to make compromises between two systems that in the end can result in suboptimal results for both.

This setup of a cognitive control constant results in a distorted view of the future states that the agent will find itself in, which becomes very costly when both systems have highly differing decision points, thereby pushing the agent to hesitate.

An interesting observation is that the agent displays some known cognitive biases that we did not explicitly model for. These cognitive biases include anchoring (i.e. relying too

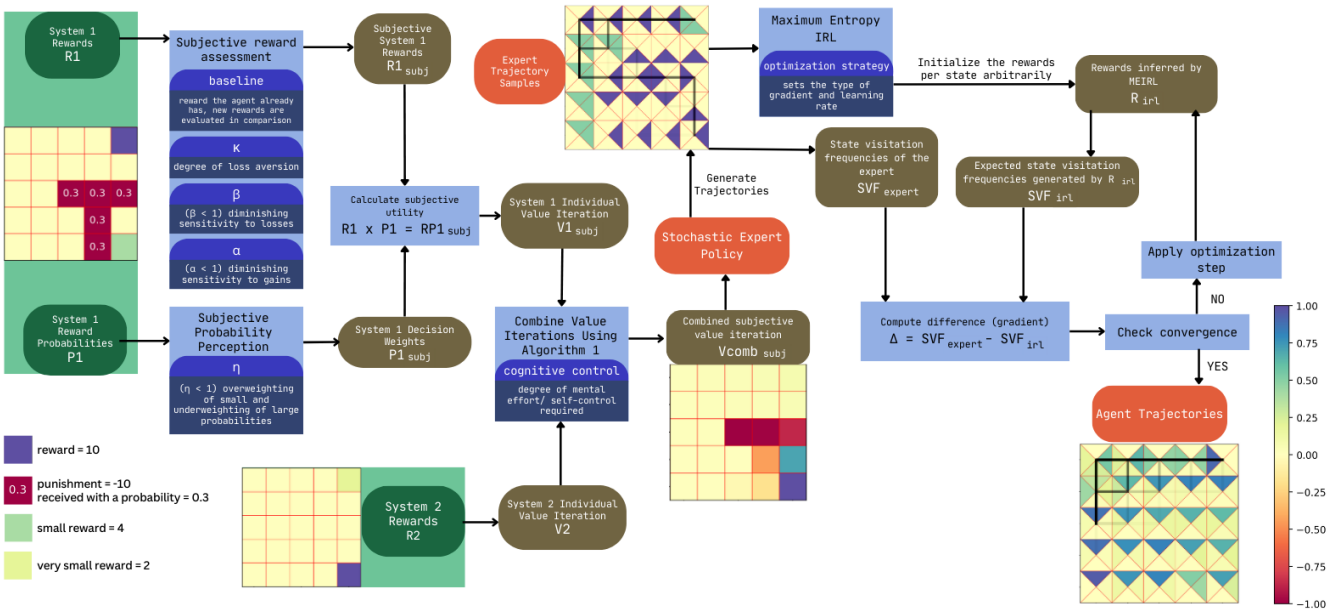

Figure 2: **Proof of Concept Experiment:** Starting From System Rewards to Generating Expert Demonstrations and Performing MEIRL (The colors are scaled for ease of reference and trajectories are visualized with their stochastic policies)

much on initial information). This is not a coincidence, but rather a result of the value iteration algorithm itself.

4.3 Experiment 1: Simple Square Grid with Bank

In this experiment the grid world pictures a small square part of the city consisting of 100 states in total, shown in figure 3 with associated R1 and R2 matrices. A total of 41 cells are road cells (possible states) and 9 of them possibly have traffic. The agent starts on the bottom left corner and has some paperwork that she needs to do in the bank (approximately middle) but she can possibly choose to skip this errand for today, go home (top right corner) and postpone visit to the bank to another day.

At the start of the episode, the agent is faced with a choice of going up or right. In contrast to Experiment 0, this road is not binding as the agent can still change course. This design mitigates the excessive effect of the first action on the rest of the episode.

The agent starts in a neutral state, with no rewards and roads are also reward neutral with the exception of the ones with traffic, indicated by red in figure 3, which incur a negative reward (punishment) to System 1 reward function with a probability. The bank has a large reward for System 1 and a small reward for System 2.Finally agent's home has a small reward for System 2 and a large reward for System 1. Applying the concepts discussed in previous sections, section 3.2, the agent evaluates the rewards of System 1 through a filter of Prospect Theory, calculating perceived (subjective) rewards and perceived probabilities called decision weights, and uses this information along with System 2 to make decisions.

This first experiment alters $R1$ and $R2$ associated with home and bank, and the magnitude and probability of negative R1 associated with busy roads, called traffic roads. The

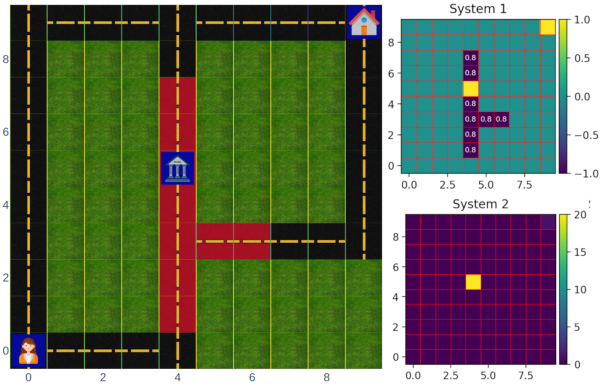

Figure 3: Visualization of Experiment 1: Square City
The values are given as an example

traffic roads are the only states that have a probability associated with them. This probability table is called $P1$. Value iteration is performed on the environment for the following cases, in the order of increasing model complexity:

- **Objective evaluation:** Take $R1 * P1 + R2$, the objective reward, as the reward function and apply conventional value iteration. This is the value iteration that the optimal agent uses to create optimal policy.
- **Two system subjective evaluation:** Apply the Prospect Theory filter to System 1 features $R1$ and $P1$, and obtain $RP1_{\text{subj}}$. Perform value iteration of System 1 and 2 separately using $R1 * P1$ and $R2$ and combine them using algorithm 1 that makes compromises between systems. This is the evaluation that expert uses to create demonstrations.

- $R_{\mathbf{irl}}$ **evaluation:** Use the expert demonstrations to infer the reward using MEIRL. Using this reward, make a conventional value iteration.

Optimal agent, Expert agent and IRL agent use the value iterations described above to make a stochastic policy and generate a set of trajectories. The average of the collected rewards, categorized by $R1$ and $R2$ are shown below in figure 4 for different values of traffic probability associated with traffic road marked red in figure 3. If both rewards are positive, they are stacked on top of each other and otherwise they both start from 0. The sums are shown in black lines. The environment and cognitive hyperparameters used in this evaluation are as follows. The home has a reward of 5 for both $R1$ and $R2$. Additionally $R2 = 30$ for the bank and $R1 = -5$ for the traffic roads. The cognitive control cost is a linear term equal to 2, representing how difficult it is to deviate from optimal actions for System 1, $\kappa = 2.0$ representing degree of loss aversion, and $\eta = 0.8$ describing the overweighting of small and underweighting of large probabilities. The terms α and β are irrelevant since the baseline (starting) reward is 0.

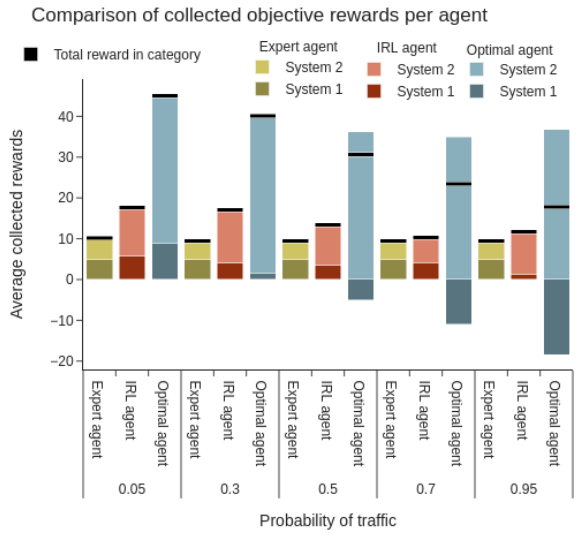

Figure 4: Rewards reached by the expert, IRL agent and completely rational optimal agent on Experiment 1

There are several factors that can cause such a significant gap between the expert demonstrations and the objective optimal. These factors include cognitive biases (loss aversion, risk aversion etc.) caused by κ and η affecting the subjective reward and decision weights respectively. As a result, the expert agent is not willing to make compromises even if this comes at the expense of further rewards (i.e. the agent is more risk averse). This can be concluded by observing the difference between sum of total rewards for the expert and optimal agent. For a small probability of a negative reward, there is a much larger reward that the expert misses.

The sum of rewards collected by the IRL agent follow the expert agent closely, especially as the probability increases. However the composition of rewards is quite different. While the expert agent keeps a balanced reward profile, the IRL agent does not differentiate, especially when the punishment is more likely. This is expected as the IRL agent does not know there are two different reward functions and can only see the end behaviour.

5 Analysis and Discussion

Using the MEIRL algorithm the IRL agent was able to infer similar paths but often failed to preserve the same balance between rewards like the expert agent does, suggesting that it was merely emulating the behaviour without inferring the motivations. Similarly, the IRL agent interprets possible outlier trajectories in the expert agent demonstrations very differently. These are expected since MEIRL is intended to derive static reward functions and not dynamically changing decision making strategies that System 1 and 2 model introduces. [10] formulates the problem as a search problem, considering multiple reward functions for different parts of the environment, where each state can be assigned a reward function and there is a probability of transition from one function to another. This can help infer that different reward functions are dominant at different parts of the environment but still does not explain the interactions of reward functions with one another.

Other papers [7] [11], also using Prospect Theory and IRL, apply mathematical techniques such as Lipschitz uniformity and continuity to derive a differentiable formula satisfying certain properties. A similar modification might make the results in this paper more consistent as well.

Qualitative analysis was critical to the experiment in this paper. The IRL algorithm used was not detailed enough to capture the nuances of the expert agent model. The expert agent model used a dynamic decision-making strategy, switching between two different reward functions; possibly with different characteristics; while the MEIRL algorithm estimates a static and singular reward function. In addition, it does not include the divergence between perceived and objective rewards. One of the solutions that we applied was limiting the number of possible states (i.e. limiting the number of roads available to the agent). After this change, the behaviour of the agent was more observable, as it was less prone to becoming stuck at non-terminal states.

5.1 Limitations

The cognitive control cost parameter was the biggest factor in the expert agent's decision making and composition of value iteration. Without further assumptions the IRL agent cannot infer the decision making process. When the cognitive control constant is 1.0 the IRL agent and expert demonstrate significantly more similar behaviour. However since cognitive control is a dynamic part of decision making in the model described in this paper, it seems unfeasible to expect a static reward guessing IRL to catch the details and intricacies involved. As a result the IRL agent is not very certain with its inferences.

Another limitation is that without actual human samples, we are only able to model known heuristics and test whether they hold ground. [13] discusses a similar problem critiquing that we model human behaviour using heuristics placing many assumptions into the problem. They consider the

question of whether it is possible to infer biases from data without further assumptions. They conclude by acknowledging that with the current resources a mix between them seem the most reasonable.

6 Responsible Research

This research examines the ability of IRL to infer rewards from expert demonstrations with cognitive biases. Since a central idea in the field of IRL is human behaviour, many ethical considerations arise. In a more extensive study, the expert demonstrations would be collected from actual humans, which creates questions about how the data was collected, stored and used. Even though the ethical implications of gathering data are not directly linked to the purposes of this paper, the implications of the research are.

Research in IRL has widespread ethical implications. By combining fields that are not conventionally within the scope of computer science, such as economics, psychology, and cognitive science, more inferences can be made about human intentions and decision-making.

The misuse of psychological analysis has been exemplified in cases like Cambridge Analytica, where personal data was used without consent to manipulate public opinion during major political events [12]. This incident raised concerns about data privacy, consent, and the potential for psychological insight from machine-learning to be used in unethical ways, such as swaying public sentiment.

Furthermore, IRL research has the capacity to simulate user's preferences based on collected user data. One of the use cases for understanding user's preferences is in social media recommendation algorithms and advertisement. While understanding the user better can lead to a more satisfactory user experience, such information also poses risks for exploitation. With a more widespread use of automated intelligent personal assistants, we want the algorithm to make decisions that align with the user's and society's long term well-being.

In an effort to promote transparency in research and deployment of IRL systems, the complete experimental setup of this paper has been made available for reproducibility and analysis in a way that is easy to design and test existing and new experiments. This approach aims to ensure accountability in IRL research. This paper uses research from non-engineering fields, such as psychology and economics, and therefore an effort is made to make the contents and results of the paper less esoteric and more accessible to researchers of other fields aside from computer science. This is a further effort to promote transparency in IRL research.

7 Conclusions and Future Work

In conclusion, while use of MEIRL has shown promise in inferring reward functions from biased expert demonstrations that demonstrate risk and loss averse behaviour, it remained an inadequate solution. Even though more sophisticated IRL methods and more complex human behaviour models that combine heuristics and data solutions might create more realistic results, this comes with the added complexity to an already computationally expensive problem. In this case, more advanced search techniques that can search a large space efficiently such as Evolutionary Algorithms would be a fruitful further area of research.

An extension of this work could be to consider individual preferences by asking personalized questions at the beginning of the learning process, in order to make better informed assumptions about the expert. This would also enhance explainability and performance in complex environments.

The experiments in this paper considered static environments with finite and known state action space. Adapting the risk sensitive cognitive biases to a continuous environment could create more realistic settings. However, again, dealing with dynamic reward systems and complex environments remains a challenge due to the rapidly increasing computational demands, as discussed previously.

Exploring alternative IRL algorithms, environments, cognitive models and large feature space search techniques could provide valuable insights into what extent can cognitive biases related to risk can be inferred using IRL. This would be beneficial due to the central role that risk plays in human decision making.

References

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

[2] Owain Evans, Andreas Stuhlmüller, and Noah D. Goodman. Learning the preferences of ignorant, inconsistent agents. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pages 323–329. AAAI Press, 2016.

[3] Meric Ikiz. Irl cognitive. https://github.com/mericikiz/irl-cognitive, 2023. Accessed on May 12, 2023.

[4] Daniel Kahneman. *Thinking, Fast and Slow*. Farrar, Straus and Giroux, 2011.

[5] Daniel Kahneman and Amos Tversky. Prospect theory: An analysis of decision under risk. *Econometrica*, 47(2):263–291, 1979.

[6] Qiang Liu. irl-maxent. GitHub repository, 2023. [Online; accessed May 12, 2023].

[7] E. Mazumdar, L.J. Ratliff, T. Fiez, and S. Shankar Sastry. Gradient-based inverse risk-sensitive reinforcement learning. In *2017 IEEE 56th Annual Conference on Decision and Control, CDC 2017*, pages 5796–5801, 2018.

[8] Andrew Y. Ng and Stuart J. Russell. Algorithms for inverse reinforcement learning. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 663–670. Morgan Kaufmann Publishers Inc., 2000.

[9] Alexander Peysakhovich. Reinforcement learning and inverse reinforcement learning with system 1 and system 2. *arXiv preprint arXiv:1805.00909*, 2018.

[10] Patrick Jaillet Quoc Phong Nguyen, Bryan Kian Hsiang Low. *Inverse Reinforcement Learning with Lo-*

cally Consistent Reward Functions. Conference on Neural Information Processing Systems, 2015.

[11] Lillian J. Ratliff and Eric Mazumdar. Inverse risk-sensitive reinforcement learning. *IEEE Transactions on Automatic Control*, 65(3):[page numbers], March 2020.

[12] Janice Richardson, Normann Witzleb, and Moira Paterson. *Big Data, Political Campaigning and the Law*. Routledge, 1st edition edition, 2019.

[13] Rohin Shah, Noah Gundotra, Pieter Abbeel, and Anca D. Dragan. On the feasibility of learning, rather than assuming, human biases for reward inference. In *Proceedings of the 37th International Conference on Machine Learning*, pages 8643–8653. PMLR, 2020.

[14] Sumeet Singh, Jonathan Lacotte, Anirudha Majumdar, and Marco Pavone. Risk-sensitive inverse reinforcement learning via semi- and non-parametric methods. *The International Journal of Robotics Research*, 37(13-14):1713–1740, 2018.

[15] Amos Tversky and Daniel Kahneman. Rational choice and the framing of decisions. *The Journal of Business*, 59(4):251–278, 1986.

[16] Peter P. Wakker. *Prospect Theory for Risk and Ambiguity*. Cambridge University Press, 2010.

[17] Brian D Ziebart. *Modeling Purposeful Adaptive Behavior with the Principle of Maximum Causal Entropy*. PhD thesis, Carnegie Mellon University, 2010.

[18] Brian D. Ziebart, Andrew Maas, J. Andrew Bagnell, and Anind K. Dey. Maximum entropy inverse reinforcement learning. In *Proc. AAAI*, pages 1433–1438, 2008.