Observational robustness for Principal-Agent games

Catalin Dumitru

February 2023

Abstract

Abstract placeholder

Contents

1 Introduction

[Paragraph about the general setting]

This project focuses on a Reinforcement Learning persuasion problem involving two parties: a principal and an agent who aim to maximize their rewards throughout the interaction. Both have access to the current state of the environment, but only the agent can take an action to change the state. However, the principal can observe an external parameter of the world, which is sampled according to a known distribution in each state. The agent doesn't have direct access to this parameter, but can receive signals from the principal and can update its belief about the parameter after receiving such signals. This parameter is important because both the princpal's and the agent's reward functions depend on it. The principal is trying to persuade the agent (through signalling) to take those actions that maximize its reward. At the same time, the agent is trying to maximize its own reward without knowing the value of the external parameter.

1.1 The setting

We can formulate this persuasion problem as an MDP defined by a tuple $\langle S, U, P, R_P, R_A, \Theta, \mu, G, \gamma_P, \gamma_A \rangle$, where

- S is a finite state space of the environment
- U is a finite action space for the agent
- $P : S \times U \to \Delta(S)$ is the transition dynamics of the state
- $R_P : S \times U \times \Theta \to \mathbb{R}$ and $R_A : S \times U \times \Theta \to \mathbb{R}$ are the reward functions for the principal and agent, respectively
- Θ is the space of the external parameter θ
- $\mu := \{\mu_s \in \Delta(\Theta) \mid s \in S\}$ is the set of distributions from which θ is drawn anew every time the state changes
- G is the space of signals for the principal
- γ_P and γ_A are the reward discount factors for the principal and agent, respectively

Both parties have access to the above information, with the only exception that the true value of θ in each state is only known by the principal.

A signalling strategy for the principal is a stochastic kernel $\pi_P : S \times \Theta \to \Delta(G)$, where $\pi_P(s, \theta, g)$ is the probability that the principal will send signal g to the agent, when the state is s and the external parameter's value is θ.

A policy for the agent is a stochastic kernel $\pi_A : S \times G \to \Delta(U)$, where $\pi_A(s, g, a)$ is the probability that the agent will take action a when the state is s and the signal received from the principal is g.

The game begins with the principal and the agent deciding on a signalling strategy π_P and a policy π_A, respectively. Then, the game proceeds in rounds: suppose the current state of the environment is s. The external parameter is drawn according to the known distribution: $\theta \sim \mu_s$. The principal observes this value and sends a signal $g \sim \pi_P(s, \theta)$ to the agent. Upon receiving the signal, the agent takes an action $a \sim \pi_A(s, g)$, the new state of the system is $s' \sim P(s, a)$ and the rewards of the principal and the agent are updated according to $R_P(s, a, \theta)$ and $R_A(s, a, \theta)$. This concludes a round, and a new one begins right after.

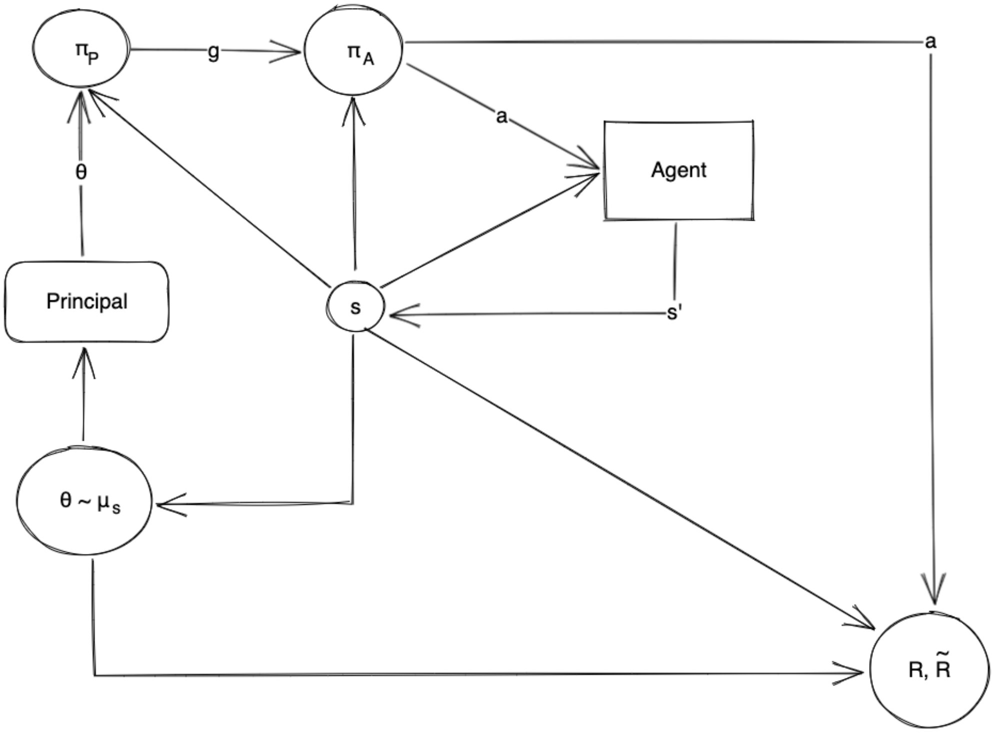

Figure 1: Interaction between different components of the game

1.2 Basic problem formulation

Next, we analyze how the problem translates to each party's point of view. Let $\mathcal{X}_P$ be the set of all signalling strategies for the principal, and $\mathcal{X}_A$ be the set of all policies for the agent.

The principal's goal is to solve the following MaxMin problem:

$$\pi_P^{\text{Opt}} := \arg\max_{\pi_P \in \mathcal{X}_P} \min_{\pi_A \in \mathcal{X}_A} \mathbb{E}\left[\sum_{k\in\mathbb{N}} \gamma_P^k R_P(s_k, a_k, \theta_k)\right] \tag{1}$$

where the expectation is taken over the trajectory $(\theta_k, g_k, a_k, s_k)_{k=0}^{\infty}$ induced by the above distributions: $\theta_k \sim \mu_{s_k}$, $g_k \sim \pi_P(s_k, \theta_k)$, $a_k \sim \pi_A(s_k, g_k)$, $s_{k+1} \sim P(s_k, a_k)$. Basically, the principal wants to find a strategy such that, whatever agent's policy he faces, the principal's worst-case reward is maximal.

Similarly, the agent's goal is to solve the following problem:

$$\pi_A^{\text{Opt}} := \arg\max_{\pi_A \in \mathcal{X}_A} \min_{\pi_P \in \mathcal{X}_P} \mathbb{E}\left[\sum_{k\in\mathbb{N}} \gamma_A^k R_A(s_k, a_k, \theta_k)\right] \tag{2}$$

where the expectation is taken over the same trajectory as above.

[Comment about the complexity of a brute-force approach? i.e. listing all possible strategies/policies]
[Add a diagram with a toy small example]

1.3 Related work

Jiarui's paper makes some additional assumptions. First, the principal announces its signalling strategy π_P to the agent at the beginning of the game. Second, the principal knows from the start what kind of agent it is dealing with. There are two main classes of agents considered in that paper:

- Myopic - only aims to maximize its immediate reward after a round
- Far-sighted - aims to maximize the overall discounted reward

Given a fixed signalling strategy of the principal, the agent can always find an optimal policy for maximizing its reward (construction in section 4.1.1 Best Response of FS Agent). Since π_P is fixed, the min in (2) goes away and the agent's problem can be solved by constructing a new MDP. (should I describe the construction, or refer the reader to Jiarui's paper?)

The agent's problem becomes

$$\pi_A^{\text{Opt}}(\pi_P) := \arg \max_{\pi_A \in \mathcal{X}_A} \mathbb{E}\left[\sum_{k\in\mathbb{N}} \gamma_A^k R_A(s_k, a_k, \theta_k)\right] \tag{3}$$

while the principal's becomes

$$\pi_P^{\text{Opt}} := \arg \max_{\substack{\pi_P \in \mathcal{X}_P \\ \pi_A = \pi_A^{\text{Opt}}(\pi_P)}} \mathbb{E}\left[\sum_{k\in\mathbb{N}} \gamma_P^k R_P(s_k, a_k, \theta_k)\right] \tag{4}$$

The principal's perspective is more complicated and interesting. It turns out that when the principal knows that the agent is myopic, an optimal signalling strategy can be computed using an additional construction of an MDP. Also, Jiarui's paper proves that it is NP-hard to find even an approximation for an optimal signalling strategy when the principal knows that the agent is far-sighted. The problem becomes tractable with some additional assumptions on the agent's behaviour.

2 Observational robustness

Inspired by the work mentioned above, this project tackles the case when the principal and the agent don't share any knowledge about the other, such as what strategy the principal commits to, or what kind of agent takes part in the game. We'll particularly focus on the agent's point of view, and we'll explore a new approach that the agent can take to try and maximize its reward without any information about the principal. First, we need some preliminary concepts and results to help us build this new class of agents, which we will call "robust", for reasons that will become apparent in the next chapter.

2.1 Preliminaries

Consider a setting where an agent observes a "noisy" state, which is different from the real state of the environment. Moreover, the agent uses this disturbed state as input for its policy. Since the agent's perception of the current state is inaccurate, the actions taken according to the policy will likely lead to sub-optimal rewards. This is where the notion of "Observational robustness" comes in. The goal is to design a policy that is "robust" against such disturbed state measurements, while also trying to obtain a reward that is as close as possible to the optimal one. Here is a formal definition of such a setting:

Definition 1 *An observationally-disturbed MDP is defined by a tuple $\langle S, U, P, R, T, \gamma \rangle$, where S, U, P, R and γ have their usual meaning, while $T : S \to \Delta(S)$ is a stochastic kernel induced by a noise signal, i.e. $T(s, s')$ is the probability of measuring state s' when the true state is actually s.*

Let $\pi : S \to \Delta(U)$ be a policy of the agent. In the classical RL setting, its value function $V^\pi : S \to \mathbb{R}$ is defined by

$$V^\pi(s_0) := \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k R(s_k, a_k)\right], \text{ where } a_k \sim \pi(s_k)$$

and we can define an objective function $J(\pi) := \mathbb{E}_{s_0 \sim \mu_0}[V^\pi(s_0)]$. The goal is to find a policy π^* such that

$$J^* := J(\pi^*) = \max_\pi J(\pi)$$

However, in the observationally-disturbed setting, the presence of T alters the policy, because the actions are applied in a potentially wrong state. Formally, the altered policy is

$$\langle \pi, T \rangle(s, a) := \sum_{s' \in S} T(s, s')\pi(s', a)$$

and we can define the "robustness regret":

$$\rho(\pi, T) := J(\pi) - J(\langle \pi, T \rangle)$$

This allows us to classify "robust" policies:

Definition 2 *A policy π is κ-robust agains a stochastic kernel T if $\rho(\pi, T) \le \kappa$.*

Finally, the goal is to minimize $J^* - J(\pi)$ as a prioritised objective and then, from the policies that satisfy this constraint, we want to pick one that is as robust as possible according to Definition 2. The point is that the disturbed policy will lead to a reward that isn't too far away from the optimal value, even with the noise present.

2.2 Related work

[Add information about Lexicographic RL and so on]

3 Robust agents

The novelty of this project's approach comes from exploiting the connection between the aforementioned papers. Going back to the principal-agent setting, we introduce the class of robust agents. We will focus on designing a policy for the agent that is robust against the external parameter θ, which acts as the disturbance or noise signal. This way, the agent also aims to be robust against the principal's signals, since those signals are designed to help the principal, not the agent.

Here is a formal setting for a robust agent against the external parameter and the principal's signals. Let

$$\mathcal{M} := \langle X, U, P, R_A, \mu, T, \gamma_A \rangle$$

be an MDP where $X := S \times G \times \Theta$ is the set of "meta-states" for the agent, $T : X \to \Delta(X)$ is a stochastic kernel, and U, P, R_A, μ and γ_A are defined as before.

Note that the only uncertainty in the agent's meta-states appears in the Θ component, since that is the external parameter only known by the principal. By definition, $T(\langle s, g, \theta \rangle, \langle s, g, \theta' \rangle)$ is the probability of the agent "measuring" value θ' for the external parameter, when the state is s, the signal received from the principal is g and the true value of the parameter is θ. Note that the agent doesn't actually "measure" anything, so we can interpret T as more of a belief or guess for the external parameter's value.

So far, I found three candidates for the kernel:

- $T(\langle s, g, \theta \rangle, \langle s, g, \theta' \rangle) := \dfrac{1}{|\Theta|}$, the uniform kernel
- $T(\langle s, g, \theta \rangle, \langle s, g, \theta' \rangle) := \mu_s(\theta')$
- $T(\langle s, g, \theta \rangle, \langle s, g, \theta' \rangle) := \dfrac{\pi_P(\theta', g)\mu_s(\theta')}{\sum_{t \in \Theta} \pi_P(t, g)\mu_s(t)}$

In the last case, since the principal doesn't share his strategy with the agent, we can use an estimate (uniform stratey, or one of the myopic/fs ones).

There are different scenarios we can explore. In all of them, the agent will always be a robust one, and we could have multiple subclasses of agents, depending on the kernel used for training. Then, every such agent can be paired with a different type of principal, depending on the signalling strategy. We have two such strategies already (myopic and far-sighted). We could also add a uniform strategy, and then we can mix them together, to test how the robust agents behave against unexpected behaviour.

This opens the door for plenty of analysis on the agent's reward. Consider the following scenario: the principal announces its strategy and promises that it would stick to it. A classic agent would just train its policy accordingly and obtain the optimal policy. However, if the principal is not trustworthy, it could fool the agent and switch its policy during the actual game, and the agent's overall reward may drop significantly. This should be less of an issue if the agent is robust, since its policy is designed to be "immune" against the principal's signals.

4 Implementation

5 Experiments

6 Conclusions

7 References