Maximum Margin IRL

1 Inverse Reinforcement Learning

Inverse Reinforcement Learning (IRL) aims to infer the underlying reward function of an agent by observing purportedly optimal behavior within a Markov Decision Process (MDP). This report details the maximum margin approach, which uses linear programming to find a reward function that optimizes a given policy while maximizing the margin between the optimal action and the next best action. Consider a finite MDP defined by the tuple (S, A, P, γ), where:

- S is a finite set of states, $|S| = N$.
- A is a finite set of actions, $|A| = K$.
- $P : S \times A \times S \to [0, 1]$ is the transition probability function, where $P(s'|s, a)$ gives the probability of transitioning to state s' from state s after taking action a.
- $\gamma \in [0, 1)$ is the discount factor.

We are given:

- An observed policy π^*, which we assume to be optimal with respect to some unknown reward function R.
- The maximum possible reward $R_{\max}$.

Our goal is to find a reward function $R : S \to [0, R_{\max}]$ such that π^* is optimal under R, and the margin between the optimal action and any other action is maximized.

2 Notation

- $V^\pi(s)$: The value function under policy π, defined as the expected cumulative discounted reward starting from state s and following policy π.
- $Q^\pi(s, a)$: The action-value function under policy π, representing the expected cumulative discounted reward starting from state s, taking action a, and thereafter following policy π.

- $\pi(s)$: The action recommended by policy π at state s.
- P_a: The state transition matrix under action a, where the entry at (s, s') is $P(s'|s, a)$.
- P_π: The state transition matrix under policy π, defined as $P_\pi(s, s') = P(s'|s, \pi(s))$.

The value function under policy π satisfies the Bellman equation:

$$V^\pi = R + \gamma P_\pi V^\pi$$

where R is the reward vector with entries $R(s)$. The critical problem is that many reward functions are usually compatible with optimal policy. Even $R(s) = 0$ everywhere. We seek a way to select one reward matrix that justifies the observed policy as optimal and is better according to some principle.

3 Maximum Margin Principle

The maximum margin principle seeks to find a reward function R that not only makes the observed policy π^* optimal but also maximizes the difference (margin) between the value of the optimal action and the next best action at each state. For each state $s \in S$, the margin $\delta(s)$ is defined as:

$$\delta(s) = \min_{a \in A, a \neq \pi^*(s)} \left[Q^{\pi^*}(s, \pi^*(s)) - Q^{\pi^*}(s, a) \right]$$

Our objective is to maximize the sum of margins over all states:

$$\text{Maximize} \quad \sum_{s \in S} \delta(s)$$

4 Linear Programming Formulation

To implement the maximum margin principle, we formulate an optimization problem as a linear program (LP).

4.1 Variables

- Reward variables: $R(s)$ for all $s \in S$.
- Margin variables: $\delta(s)$ for all $s \in S$.

4.2 Objective Function

We aim to maximize the total margin while applying regularization to the rewards:

$$\text{Maximize} \quad \sum_{s \in S} \delta(s) - \lambda \sum_{s \in S} R(s)$$

where $\lambda \geq 0$ is the regularization parameter.

4.3 Constraints

1. **Policy Optimality Constraints:**

   For all $s \in S$ and $a \in A$, we require that the observed policy π^* is optimal under R:

   $$(P_{\pi^*}(s,:) - P_a(s,:))(I - \gamma P_{\pi^*})^{-1} R \geq 0, \quad \forall a \neq \pi^*(s)$$

2. **Margin Constraints:**

   For all $s \in S$ and $a \in A$:

   $$\delta(s) \leq (P_{\pi^*}(s,:) - P_a(s,:))(I - \gamma P_{\pi^*})^{-1} R, \quad \forall a \neq \pi^*(s)$$

3. **Reward Bounds:**

   $$0 \leq R(s) \leq R_{\max}, \quad \forall s \in S$$

4. **Non-negativity of Margins:**

   $$\delta(s) \geq 0, \quad \forall s \in S$$

4.4 Canonical LP Form

We can express the LP in the standard canonical form:

$$\begin{aligned} &\text{Maximize} \quad c^T x \\ &\text{Subject to} \quad Ax \leq b \\ &\quad x \geq 0 \end{aligned}$$

where:

- x is the vector of variables, consisting of $\delta(s)$ and $R(s)$ for all $s \in S$.
- c is the objective coefficients vector.
- A and b define the inequality constraints.

5 Example: 2-State, 2-Action MDP

Consider a simple Markov Decision Process (MDP) with the following specifications:

- **States:** $S = \{s_0, s_1\}$
- **Actions:** $A = \{a_0, a_1\}$
- **Transition Probabilities:**

$$P_{a_0} = \begin{pmatrix} 0.5501 & 0.5000 \\ 0.0628 & 0.4591 \end{pmatrix}, \quad P_{a_1} = \begin{pmatrix} 0.5000 & 0.5000 \\ 0.4592 & 0.5408 \end{pmatrix}$$

- **Discount Factor:** $\gamma = 0.9$
- **Maximum Reward:** $R_{\max} = 10$
- **Observed Optimal Policy:**

$$\pi^*(s_0) = a_1, \quad \pi^*(s_1) = a_0$$

5.1 Constructing the Linear Program

The goal is to infer the reward function $R = \begin{pmatrix} R(s_0) \\ R(s_1) \end{pmatrix}$ that justifies the observed optimal policy π^* while maximizing the margin between the optimal and suboptimal actions.

5.1.1 Compute P_{π^*} and $(I - \gamma P_{\pi^*})^{-1}$

First, construct the transition matrix under the optimal policy π^*:

$$P_{\pi^*} = \begin{pmatrix} P_{\pi^*}(s_0 \to s') \\ P_{\pi^*}(s_1 \to s') \end{pmatrix} = \begin{pmatrix} P_{a_1}(s_0 \to s') \\ P_{a_0}(s_1 \to s') \end{pmatrix} = \begin{pmatrix} 0.5000 & 0.5000 \\ 0.0628 & 0.4591 \end{pmatrix}$$

Next, compute the matrix $A = I - \gamma P_{\pi^*}$:

$$A = I - \gamma P_{\pi^*} = \begin{pmatrix} 1 - 0.9 \times 0.5000 & -0.9 \times 0.5000 \\ -0.9 \times 0.0628 & 1 - 0.9 \times 0.4591 \end{pmatrix} = \begin{pmatrix} 0.5500 & -0.4500 \\ -0.0565 & 0.5862 \end{pmatrix}$$

Compute the inverse of A:

$$A^{-1} = \begin{pmatrix} 5.1351 & 4.8649 \\ 2.4324 & 7.5676 \end{pmatrix}$$

5.1.2 Compute $M_a(s)$ for $a \neq \pi^*(s)$

For each state s and action $a \neq \pi^*(s)$, compute:

$$M_a(s, :) = (P_{\pi^*}(s, :) - P_a(s, :))\, A^{-1}$$

State s_0, Action a_0

$$\Delta P(s_0) = P_{\pi^*}(s_0,:) - P_{a_0}(s_0,:) = \begin{pmatrix} 0.5000 - 0.5501 & 0.5000 - 0.5000 \end{pmatrix} = \begin{pmatrix} -0.0501 & 0.0000 \end{pmatrix}$$

$$M_{a_0}(s_0,:) = \Delta P(s_0) \cdot A^{-1} = \begin{pmatrix} -0.0501 & 0.0000 \end{pmatrix} \begin{pmatrix} 5.1351 & 4.8649 \\ 2.4324 & 7.5676 \end{pmatrix} = \begin{pmatrix} -0.2570 & -0.2438 \end{pmatrix}$$

State s_1, Action a_1

$$\Delta P(s_1) = P_{\pi^*}(s_1,:) - P_{a_1}(s_1,:) = \begin{pmatrix} 0.0628 - 0.4592 & 0.4591 - 0.5408 \end{pmatrix} = \begin{pmatrix} -0.3964 & -0.0817 \end{pmatrix}$$

$$M_{a_1}(s_1,:) = \Delta P(s_1) \cdot A^{-1} = \begin{pmatrix} -0.3964 & -0.0817 \end{pmatrix} \begin{pmatrix} 5.1351 & 4.8649 \\ 2.4324 & 7.5676 \end{pmatrix} = \begin{pmatrix} -4.6929 & -4.8649 \end{pmatrix}$$

5.1.3 Formulating the Constraints

Define the variable vector:

$$x = \begin{pmatrix} \delta(s_0) \\ \delta(s_1) \\ R(s_0) \\ R(s_1) \end{pmatrix}$$

Define the objective function coefficients:

$$c = \begin{pmatrix} -1 \\ -1 \\ 0.1 \\ 0.1 \end{pmatrix}$$

Construct the inequality constraint matrix A and vector b based on the constraints.

Margin Constraints For each state s and action $a \neq \pi^*(s)$:

$$\delta(s) \leq M_a(s,:)R$$

This translates to:

$$\begin{cases} \delta(s_0) - (-0.2570R(s_0) - 0.2438R(s_1)) \leq 0 \\ \delta(s_1) - (-4.6929R(s_0) - 4.8649R(s_1)) \leq 0 \end{cases}$$

Which simplifies to:

$$\begin{cases} \delta(s_0) + 0.2570R(s_0) + 0.2438R(s_1) \leq 0 \\ \delta(s_1) + 4.6929R(s_0) + 4.8649R(s_1) \leq 0 \end{cases}$$

Policy Optimality Constraints For each state s and action $a \neq \pi^*(s)$:

$$M_a(s,:)R \geq 0$$

This translates to:

$$\begin{cases} -0.2570R(s_0) - 0.2438R(s_1) \geq 0 \\ -4.6929R(s_0) - 4.8649R(s_1) \geq 0 \end{cases}$$

Which simplifies to:

$$\begin{cases} 0.2570R(s_0) + 0.2438R(s_1) \leq 0 \\ 4.6929R(s_0) + 4.8649R(s_1) \leq 0 \end{cases}$$

Reward Bounds For each state s:

$$0 \leq R(s) \leq 10$$

Non-negativity of Margins For each state s:

$$\delta(s) \geq 0$$

5.2 Final LP Formulation

Summarizing the constructed LP:

$$\begin{aligned} \text{Minimize} \quad & -\delta(s_0) - \delta(s_1) + 0.1R(s_0) + 0.1R(s_1) \\ \text{Subject to} \quad & \begin{cases} \delta(s_0) + 0.2570R(s_0) + 0.2438R(s_1) \leq 0 \\ \delta(s_1) + 4.6929R(s_0) + 4.8649R(s_1) \leq 0 \\ -0.2570R(s_0) - 0.2438R(s_1) \leq 0 \\ -4.6929R(s_0) - 4.8649R(s_1) \leq 0 \\ R(s_0) \leq 10 \\ R(s_1) \leq 10 \end{cases} \\ & \delta(s_0) \geq 0, \quad \delta(s_1) \geq 0 \\ & R(s_0) \geq 0, \quad R(s_1) \geq 0 \end{aligned}$$

In matrix form:

$$A = \begin{pmatrix} 1 & 0 & 0.2570 & 0.2438 \\ 0 & 1 & 4.6929 & 4.8649 \\ 0 & 0 & -0.2570 & -0.2438 \\ 0 & 0 & -4.6929 & -4.8649 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix}, \quad b = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 10 \\ 10 \\ 0 \\ 0 \end{pmatrix}$$

$$c = \begin{pmatrix} -1 \\ -1 \\ 0.1 \\ 0.1 \end{pmatrix}$$

$$x = \begin{pmatrix} \delta(s_0) \\ \delta(s_1) \\ R(s_0) \\ R(s_1) \end{pmatrix}$$

The solution to this LP provides the estimated reward function $R_{\text{est}} = \begin{pmatrix} 0 \\ 10 \end{pmatrix}$ and margins $\delta_{\text{est}} = \begin{pmatrix} 0 \\ 10 \end{pmatrix}$, which justify the observed optimal policy π^*.