Inverse Reinforcement Learning

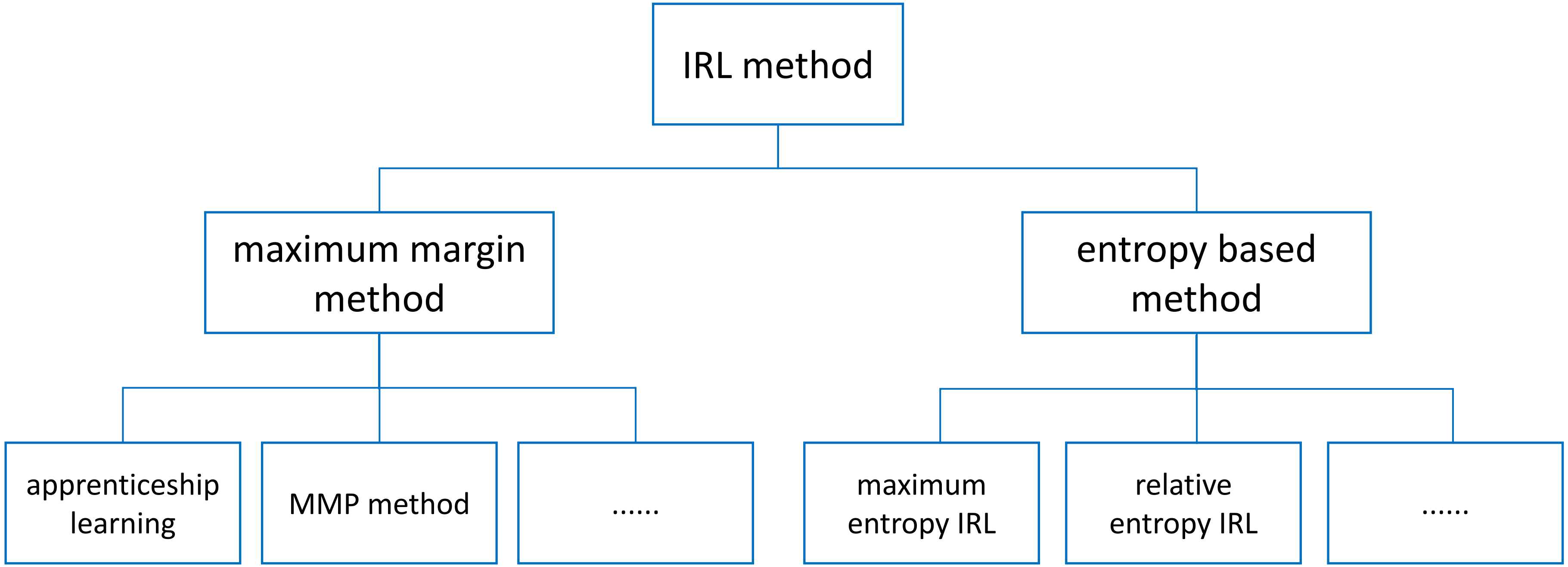
presented by Jason TOKO

Inverse Reinforcement Learning

- What is IRL?
  - IRL is a part of IL (maybe)
  - IRL learns from an expert to recover unknown reward function(**R***)

- Why we use IRL?
  - Sometimes it may be difficult to construct explicit reward function
  - Or the given reward function

Inverse Reinforcement Learning



Apprenticeship Learning

- Some preliminaries:
  - feature vector: $\phi(s)$
  - reward function(linear): $R(s) = w \cdot \phi(s)$
  - value function of π : $E_{s_0 \sim D}[V^{\pi}(s_0)] = E[\sum_{t=0}^{\infty} \gamma^t R(s_t) \mid \pi]$

$$= E[\sum_{t=0}^{\infty} \gamma^t w \cdot \phi(s_t) \mid \pi]$$

$$= w \cdot E[\sum_{t=0}^{\infty} \gamma^t \phi(s_t) \mid \pi]$$

$$= w \cdot \mu(\pi)$$

  - thus,we get feature expectations: $\mu(\pi) = E[\sum_{t=0}^{\infty} \gamma^t \phi(s_t) \mid \pi]$

Apprenticeship Learning

- Some preliminaries:
  - expert's trajectories: $\{s_0^{(i)}, s_1^{(i)}, s_2^{(i)}, \ldots\}_{i=1}^{m}$
  - expert's feature expectations: $\mu_E = \mu(\pi_E) = \frac{1}{m}\sum_{i=1}^{m}\sum_{t=0}^{\infty}\gamma^t \phi(s_t^{(i)})$

- Problem formulation:
  - given : MDP, $\phi(s)$, μ_E
  - to find : $\tilde{\pi}$ such that $\| \mu(\tilde{\pi}) - \mu_E \|_2 \leq \varepsilon$

Apprenticeship Learning

- Algorithm:

(pseudocode in the paper)

1. Randomly pick some policy $\pi^{(0)}$, compute (or approximate via Monte Carlo) $\mu^{(0)} = \mu(\pi^{(0)})$, and set i = 1.

2. Compute $t^{(i)} = \max_{w:\|w\|_2 \le 1} \min_{j \in \{0..i-1\}} w^T(\mu_E - \mu^{(j)})$, and let $w^{(i)}$ be the value of w that attains this maximum.

3. If $t^{(i)} \le \varepsilon$, then terminate.

4. Using the RL algorithm, compute the optimal policy $\pi^{(i)}$ for the MDP using rewards $R = (w^{(i)})^T \phi$

5. Compute (or estimate) $\mu^{(i)} = \mu(\pi^{(i)})$.

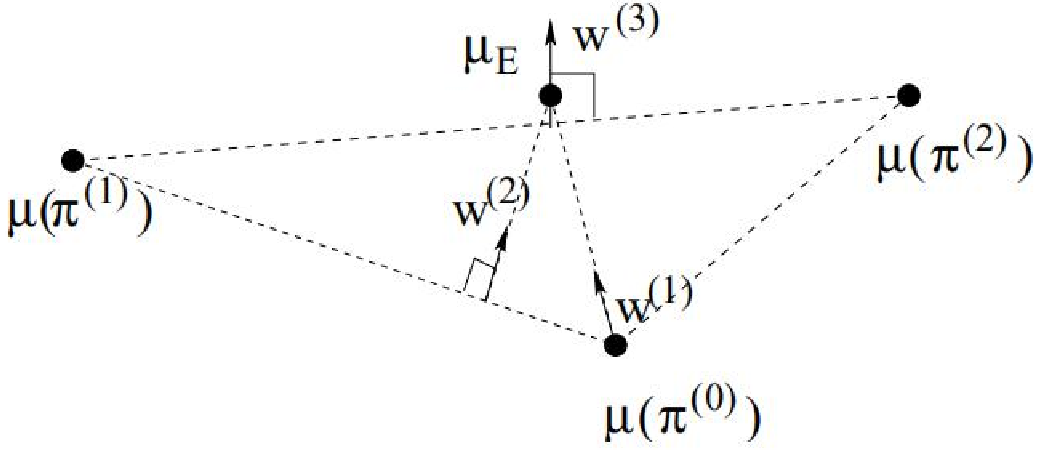
6. Set i = i + 1, and go back to step 2.

Upon termination, the algorithm returns $\{\pi^{(i)} : \mathrm{i} = 0 \ldots \mathrm{n}\}$

Apprenticeship Learning

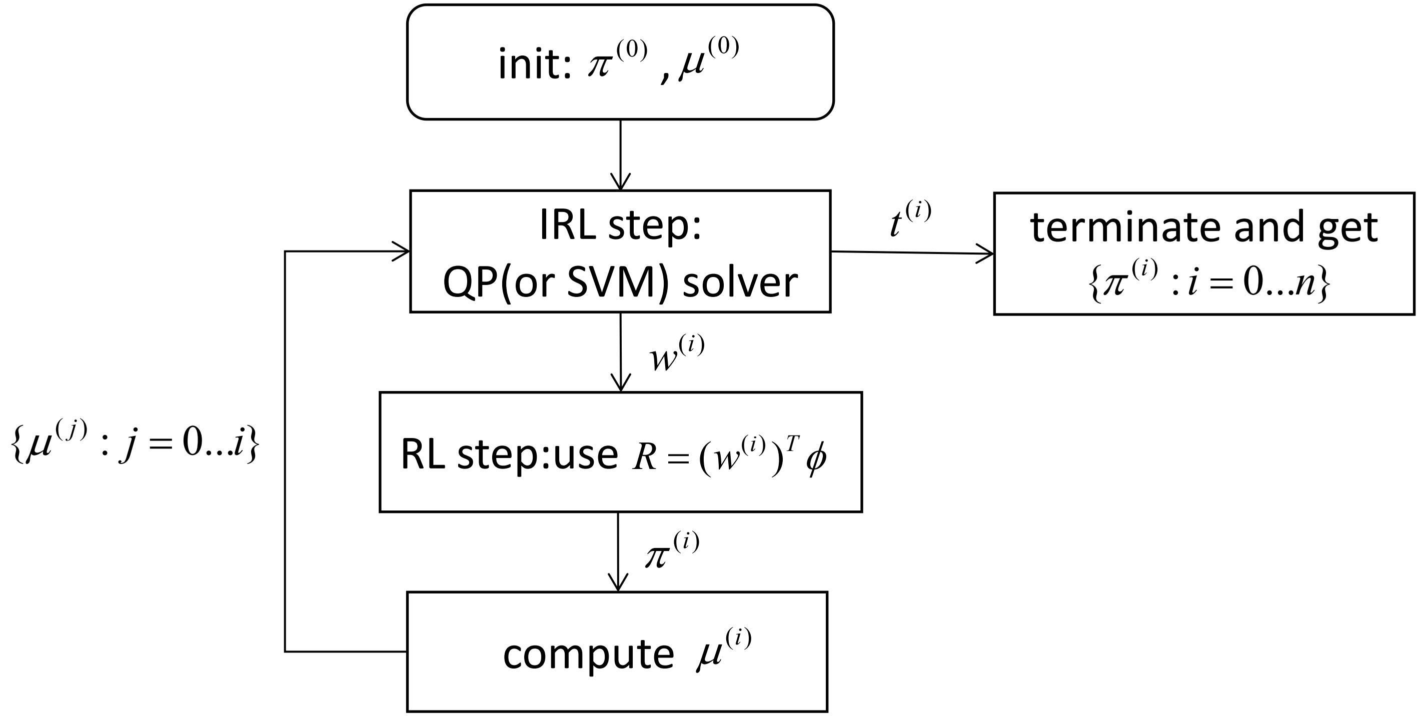
On the last page, the optimization in step 2 can be equivalently written as:

$$\begin{aligned} \max_{t,w} \quad & t \\ \text{s.t.} \quad & w^T \mu_E \geq w^T \mu^{(j)} + t,\ j = 0, \ldots, i-1 \\ & ||w||_2 \leq 1 \end{aligned}$$



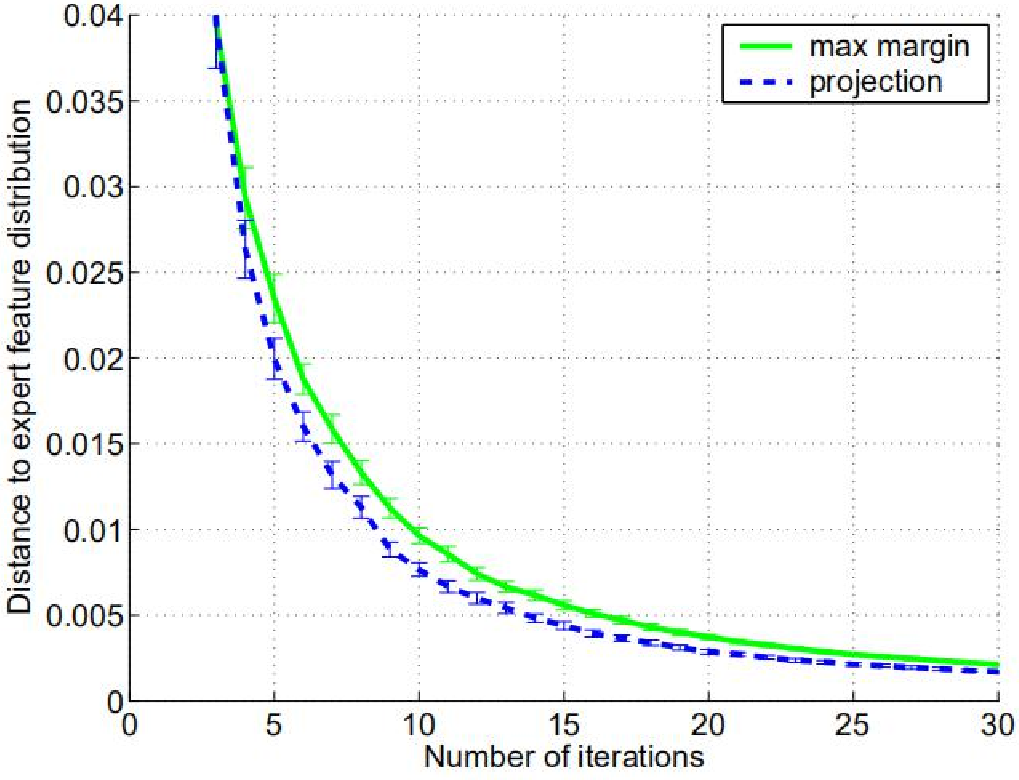
WOW!! Under this form, we can use SVM to find the $w^{(i)}$!

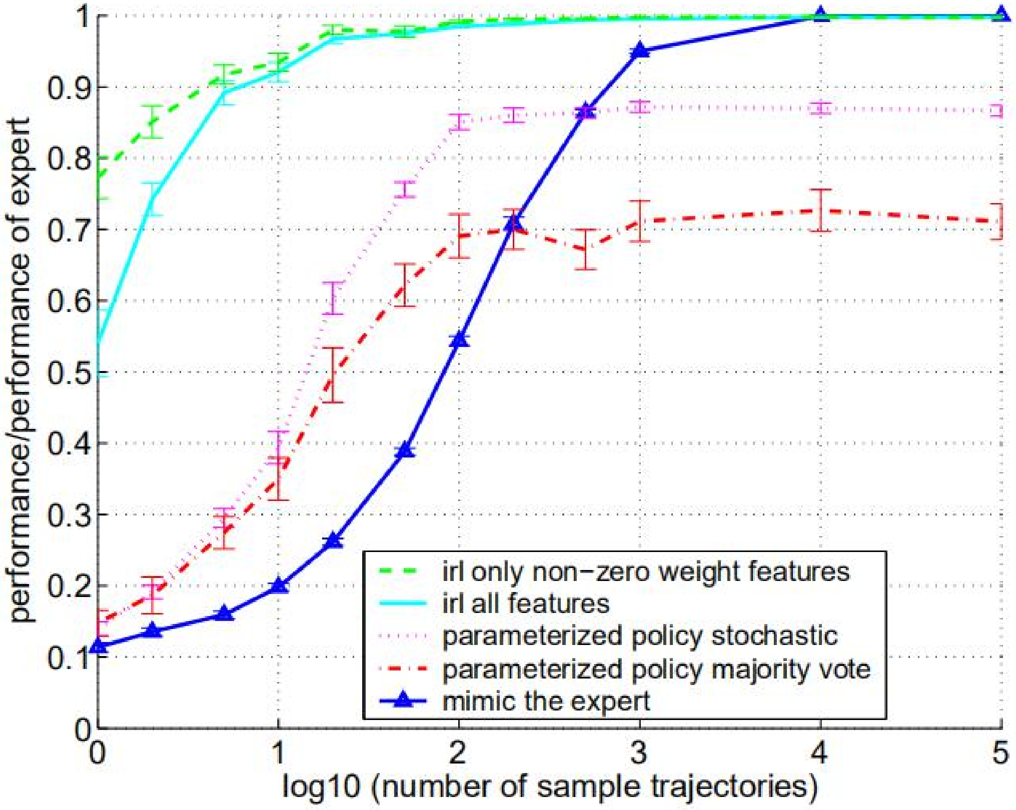
Apprenticeship Learning



Experiments

- Gridworld





Conclusion

- "Even though we cannot guarantee that our algorithms will correctly recover the expert's true reward function, we show that our algorithm will nonetheless find **a policy that performs as well as the expert**, where performance is measured with respect to the expert's unknown reward function." ------from "Apprenticeship Learning via Inverse Reinforcement Learning"
- That means maximum margin methods may lead to ambiguity: maximum entropy method can solve it!(BUT I WILL NOT PRESENT HERE !HHHHH!)