Semi-Supervised Inverse Reinforcement Learning

Joint work with Mohammad Ghavamzadeh and Alessandro Lazaric
Motivation (Apprenticeship Learning)

• **Traditional Reinforcement Learning (RL)**
  - Reward algorithms for being in certain states
  - Takes lot of experts’ time (human knowledge)
  - Difficult to encode

• **Apprenticeship Learning (Inverse RL)**
  - Input: Behavior = experts’ trajectories
  - Find a policy that resembles the expert’s
  - Find a reward for which is the behavior optimal
Successes of Apprenticeship Learning

heli.stanford.edu

Kolter et al. NIPS 2008

Ziebart et al. AAAI 2008
Motivation (Semi-Supervised AL)

- **Main motivation: reduce humans’ effort**
  - Encoding the reward function
  - Demonstration of good behavior
- **RL vs. AL:**
  - reward function
  - demonstrations
- **AL vs. SSAL:**
  - only expert’s trajectories
  - expert’s + *unlabeled* trajectories
Semi-Supervised Inverse RL

unlabeled trajectories → imitation learning → labeled trajectories → inverse RL → reward → reinforcement learning → policy → semi-supervised inverse RL
Advantages of the setting

• Apprenticeship learning
  ▪ May require many experts’ trajectories
  ▪ Expert trajectories can be costly to get

• Semi-supervised apprenticeship learning
  ▪ (non-expert) trajectories could be available
  ▪ Examples: online gaming, cheap learning

Goal: reduce #expert trajectories or speed up learning (fewer iterations)
Approaches

• Apprenticeship Learning via Inverse Reinforcement Learning
  ▪ Abbeel, Ng, ICML 2004

• Maximum Entropy Inverse RL
  ▪ Ziebart, Maas, Bagnell, Dey, AAAI 2008

• Max-Margin Planning
  ▪ Ratliff, Bagnell, Zinkevich, ICML 2006

• IRL via Reduction to Classification
  ▪ Syed, Shapire, NIPS 2010
  ▪ Ross, Bagnell, AISTATS 2010

• Inverse Optimal Control with Linearly Solvable MDPs
  ▪ Dvijotham, Todorov, ICML 2010
AR via IRL (Abbeel & Ng, 2004)

- Reward is linear in features defined over the states

\[ R^*(s) = \mathbf{w}^* \cdot \phi(s) \]

- Expected value of the policy:

\[
\mathbb{E}_{s_0 \sim D}[V^\pi(s_0)] = \mathbb{E} \left[ \sum_{t=0}^{\infty} \gamma^t R(s_t) | \pi \right] = \mathbf{w} \cdot \mathbb{E} \left[ \sum_{t=0}^{\infty} \gamma^t \phi(s_t) | \pi \right] = \mathbf{w} \cdot \mu(\pi)
\]

- Find policy matching expert’s feature counts:

\[
\left| \mathbb{E} \left[ \sum_{t=0}^{\infty} \gamma^t R(s_t) | \pi_E \right] - \mathbb{E} \left[ \sum_{t=0}^{\infty} \gamma^t R(s_t) | \tilde{\pi} \right] \right| = |\mathbf{w}^T \mu(\tilde{\pi}) - \mathbf{w}^T \mu_E| \leq ||\mathbf{w}||_2 \|\mu(\tilde{\pi}) - \mu_E\|_2 \leq \varepsilon
\]
Original IRL Algorithm (max-margin version)

SVM classification
Cluster assumption for semi-supervised SVMs

only labeled data   with unlabeled data
**SSIRL algorithm**

**Input:** \( \varepsilon, \gamma_l, \gamma_u \)
- expert trajectories \( \{ s^{(i)}_{E,t} \} \)
- unlabeled trajectories from \( U \) performers \( \{ s^{(i)}_{u,t} \} \)

**Estimate**
- \( \hat{\mu}_E \leftarrow \frac{1}{m} \sum_{i=1}^{m} \sum_{t=0}^{\infty} \gamma_l^t \phi(s^{(i)}_{E,t}) \)
- for \( u = 1 \) to \( U \) do
  - estimate \( \hat{\mu}_u \leftarrow \frac{1}{m_u} \sum_{i=1}^{m_u} \sum_{t=0}^{\infty} \gamma_l^t \phi(s^{(i)}_{u,t}) \)
end for

**Randomly** pick \( \pi^{(0)} \) and set \( i \leftarrow 1 \)

**Repeat**
- \( w^{(i)} \leftarrow \min_w \left( \max\{1 - w^T \hat{\mu}_E, 0\} \right. \)
- \( + \gamma_l \|w\|_2 + \sum_{j<i} \max\{1 + w^T \hat{\mu}^{(j)}, 0\} \)
- \( + \gamma_u \sum_{u \in U} \max\{1 - \|w^T \hat{\mu}_u\|, 0\} \)

- \( w^{(i)} \leftarrow w^{(i)}/\|w^{(i)}\|_2 \)
- \( \pi^{(i)} \leftarrow \text{MDP}(R = (w^{(i)})^T \phi) \)
- estimate \( \hat{\mu}^{(i)} \leftarrow \mu(\pi^{(i)}) \)
- \( t^{(i)} \leftarrow \min_i w^T(\hat{\mu}_E - \hat{\mu}^{(i)}) \)
- \( i \leftarrow i + 1 \)

**Until** \( t^{(i)} \leq \varepsilon \)
Grid world experiments

- same setup as Abbeel and Ng (2004)
- with vs. without unlabeled trajectories
- 64 x 64 gridworlds
- 4 actions (north, west, south, east)
- 70% of success and 30% different action
- 64 features: 8 x 8 macrocells
Experimental setup
Advantage of unlabeled data
Convergence of the SSIRL algorithm
Discussion

• **Contributions:**
  - **first** IRL method that uses **unlabeled** trajectories
  - assuming **clustered** feature counts can learn a better performing policy

• **Disadvantages:**
  - similar to Abbeel and Ng (2004) only outputs a **mixture** policy
  - stopping criterion is needed, because the method **converges to IRL** of Abbeel and Ng (2004)
Discussion

• Open questions:
  ▪ Do real-world problems satisfy distributional assumptions that we can leverage?
  ▪ For which tasks can we obtain « cheap » trajectories?

• Future directions:
  ▪ enhance other inverse RL methods (MaxEnt IRL, MMP, ... ) with unlabeled trajectories
  ▪ investigate manifold assumption for inverse RL