Behavioral cloning example

- state: board configuration + shape of the falling piece \( \sim 2^{200} \) states!
- action: rotation and translation applied to the falling piece

Behavioral cloning example

- 22 features aka basis functions \( \phi_i \):
  - Ten basis functions, 0, \ldots, 9, mapping the state to the height \( h[k] \) of each of the ten columns.
  - Nine basis functions, 10, \ldots, 18, each mapping the state to the absolute difference between heights of successive columns: \( |h[k+1] - h[k]| \), \( k = 1, \ldots, 9 \).
  - One basis function, 19, that maps state to the maximum column height: \( \max_k h[k] \)
  - One basis function, 20, that maps state to the number of ‘holes’ in the board.
  - One basis function, 21, that is equal to 1 in every state.

\[
V(s) = \sum_{i=1}^{22} \theta_i \phi_i(s)
\]

[Bertsekas & Ioffe, 1996 (TD); Bertsekas & Tsitsiklis, 1996 (TD); Kakade, 2002 (policy gradient); Farias & Van Roy, 2006 (approximate LP)]
Behavioral cloning example
**Behavioral cloning example**

**Training data:** Example choices of next states chosen by the demonstrator: $s^{(i)}_k$
Alternative choices of next states that were available: $s^{(i)}_{j-}$

Max-margin formulation

$$\min_{\theta, \xi \geq 0} \quad \theta^\top \theta + C \sum_{i,j} \xi_{i,j}$$

subject to \( \forall i, \forall j : \theta^\top \phi(s^{(i)}_k) \geq \theta^\top \phi(s^{(i)}_{j-}) + 1 - \xi_{i,j} \)

Probabilistic/Logistic formulation

Assumes experts choose for result $s^{(i)}$ with probability

$$\frac{\exp(\theta^\top \phi(s^{(i)}_k))}{\exp(\theta^\top \phi(s^{(i)}_k) + \sum_{j-} \exp(\theta^\top \phi(s^{(i)}_{j-})))}.$$

Hence the maximum likelihood estimate is given by:

$$\max_{\theta} \sum_i \log \left( \frac{\exp(\theta^\top \phi(s^{(i)}_k))}{\exp(\theta^\top \phi(s^{(i)}_k) + \sum_{j-} \exp(\theta^\top \phi(s^{(i)}_{j-})))} \right) - C \|\theta\|$$

---

**Motivation for inverse RL**

- **Scientific inquiry**
  - Model animal and human behavior
    - E.g., bee foraging, songbird vocalization. [See intro of Ng and Russell, 2000 for a brief overview.]
  
- **Apprenticeship learning/Imitation learning through inverse RL**
  - Presupposition: reward function provides the most succinct and transferable definition of the task
  - Has enabled advancing the state of the art in various robotic domains
  
- **Modeling of other agents, both adversarial and cooperative**
Problem setup

Input:
- State space, action space
- Transition model $P_{sa}(s_{t+1} | s_t, a_t)$
- No reward function
- Teacher’s demonstration: $s_0, a_0, s_1, a_1, s_2, a_2, ...$ (= trace of the teacher’s policy $\pi^*$)

- Inverse RL:
  - Can we recover $R$?

- Apprenticeship learning via inverse RL
  - Can we then use this $R$ to find a good policy?

- Vs. Behavioral cloning (which directly learns the teacher’s policy using supervised learning)
  - Inverse RL: leverages compactness of the reward function
  - Behavioral cloning: leverages compactness of the policy class considered, does not require a dynamics model

Lecture outline

- Inverse RL intro
- **Mathematical formulations for inverse RL**
- Case studies
Three broad categories of formalizations

- Max margin
- Feature expectation matching
- Interpret reward function as parameterization of a policy class

Basic principle

- Find a reward function $R^*$ which explains the expert behaviour.
- Find $R^*$ such that $E[\sum_{t=0}^{\infty} \gamma^t R^*(s_t)|\pi^*] \geq E[\sum_{t=0}^{\infty} \gamma^t R^*(s_t)|\pi] \quad \forall \pi$

Equivalently, find $R^*$ such that

$$\sum_{s \in S} R(s) \left( \sum_{t=0}^{\infty} \gamma^t 1\{s_t = s|\pi^*\} \right) \geq \sum_{s \in S} R(s) \left( \sum_{t=0}^{\infty} \gamma^t 1\{s_t = s|\pi^*\} \right) \quad \forall \pi$$

A convex feasibility problem in $R^*$, but many challenges:
- $R=0$ is a solution, more generally: reward function ambiguity
- We typically only observe expert traces rather than the entire expert policy $\pi^*$ --- how to compute LHS?
- Assumes the expert is indeed optimal --- otherwise infeasible
- Computationally: assumes we can enumerate all policies
Feature based reward function

- Let $R(s) = w^\top \phi(s)$, where $w \in \mathbb{R}^n$, and $\phi : S \to \mathbb{R}^n$.

$$E[ \sum_{t=0}^{\infty} \gamma^t R(s_t) | \pi ] =$$

---

Feature based reward function

- Let $R(s) = w^\top \phi(s)$, where $w \in \mathbb{R}^n$, and $\phi : S \to \mathbb{R}^n$.

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

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

$$= w^\top \mu(\pi)$$

Expected cumulative discounted sum of feature values or “feature expectations”

- Subbing into $E[ \sum_{t=0}^{\infty} \gamma^t R^*(s_t) | \pi^* ] \geq E[ \sum_{t=0}^{\infty} \gamma^t R^*(s_t) | \pi ] \quad \forall \pi$
gives us:

Find $w^*$ such that $w^*^\top \mu(\pi^*) \geq w^*^\top \mu(\pi) \quad \forall \pi$
Feature based reward function

Let $R(s) = w^T \phi(s)$, where $w \in \mathbb{R}^n$, and $\phi : S \rightarrow \mathbb{R}^n$.
Find $w^*$ such that $w^* \mu(\pi^*) \geq w^* \mu(\pi) \quad \forall \pi$

- Feature expectations can be readily estimated from sample trajectories.
- The number of expert demonstrations required scales with the number of features in the reward function.
- The number of expert demonstration required does not depend on
  - Complexity of the expert’s optimal policy $\pi^*$
  - Size of the state space

Recap of challenges

Let $R(s) = w^T \phi(s)$, where $w \in \mathbb{R}^n$, and $\phi : S \rightarrow \mathbb{R}^n$.
Find $w^*$ such that $w^* \mu(\pi^*) \geq w^* \mu(\pi) \quad \forall \pi$

- Challenges:
  - Assumes we know the entire expert policy $\pi^*$ $\rightarrow$ assumes we can estimate expert feature expectations
  - $R=0$ is a solution (now: $w=0$), more generally: reward function ambiguity
  - Assumes the expert is indeed optimal--became even more of an issue with the more limited reward function expressiveness!
  - Computationally: assumes we can enumerate all policies
Ambiguity

- We currently have: Find $w^*$ such that $w^* \mu(\pi^*) \geq w^* \mu(\pi) \quad \forall \pi$

- Standard max margin:

- “Structured prediction” max margin:

\[
\min_w \|w\|^2_2 \\
\text{s.t. } w^T \mu(\pi^*) \geq w^T \mu(\pi) + 1 \\
\forall \pi
\]

“Structured prediction” max margin:

\[
\min_w \|w\|^2_2 \\
\text{s.t. } w^T \mu(\pi^*) \geq w^T \mu(\pi) + m(\pi^*, \pi) \\
\forall \pi
\]

Justification: margin should be larger for policies that are very different from $\pi^*$.

Example: $m(\pi, \pi^*) = \text{number of states in which } \pi^* \text{ was observed and in which } \pi \text{ and } \pi^* \text{ disagree}$
Expert suboptimality

- Structured prediction max margin:
  \[
  \min_w \|w\|_2^2 \\
  \text{s.t. } w^\top \mu(\pi^*) \geq w^\top \mu(\pi) + m(\pi^*, \pi) \quad \forall \pi
  \]

- Structured prediction max margin with slack variables:
  \[
  \min_{w, \xi} \|w\|_2^2 + C \xi \\
  \text{s.t. } w^\top \mu(\pi^*) \geq w^\top \mu(\pi) + m(\pi^*, \pi) - \xi \quad \forall \pi
  \]

- Can be generalized to multiple MDPs (could also be same MDP with different initial state)
  \[
  \min_{w, \xi^{(i)}} \|w\|_2^2 + C \sum_i \xi^{(i)} \\
  \text{s.t. } w^\top \mu(\pi^{(i)*}) \geq w^\top \mu(\pi^{(i)}) + m(\pi^{(i)*}, \pi^{(i)}) - \xi^{(i)} \quad \forall i, \pi^{(i)}
  \]
Resolved: access to $\pi^*$, ambiguity, expert suboptimality

One challenge remains: very large number of constraints

- Ratliff+al use subgradient methods.
- In this lecture: constraint generation

Complete max-margin formulation

$$\begin{align*}
\min_w & \quad \|w\|_2^2 + C \sum_i \xi^{(i)} \\
\text{s.t.} & \quad w^T \mu(\pi^{(i)*}) \geq w^T \mu(\pi^{(i)}) + m(\pi^{(i)*}, \pi^{(i)}) \quad \forall i, \pi^{(i)}
\end{align*}$$

[Ratliff, Zinkevich and Bagnell, 2006]

Constraint generation

Initialize $\Pi^{(i)} = \emptyset$ for all $i$ and then iterate

- Solve

$$\begin{align*}
\min_w & \quad \|w\|_2^2 + C \sum_i \xi^{(i)} \\
\text{s.t.} & \quad w^T \mu(\pi^{(i)*}) \geq w^T \mu(\pi^{(i)}) + m(\pi^{(i)*}, \pi^{(i)}) - \xi^{(i)} \quad \forall i, \forall \pi^{(i)} \in \Pi^{(i)}
\end{align*}$$

- For current value of $w$, find the most violated constraint for all $i$ by solving:

$$\max_{\pi^{(i)}} w^T \mu(\pi^{(i)}) + m(\pi^{(i)*}, \pi^{(i)})$$

= find the optimal policy for the current estimate of the reward function (+ loss augmentation $m$)

- For all $i$ add $\pi^{(i)}$ to $\Pi^{(i)}$

- If no constraint violations were found, we are done.
Every policy $\pi$ has a corresponding feature expectation vector $\mu(\pi)$, which for visualization purposes we assume to be 2D.

Visualization in feature expectation space

- $\mu(\pi_*)$
- $\mu(\pi_1)$
- $\mu(\pi_2)$

$\max_{\pi} \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t) \mid \pi\right] = w^T \mu(\pi)$

Max margin

Structured max margin

Constraint generation

- Every policy $\pi$ has a corresponding feature expectation vector $\mu(\pi)$, which for visualization purposes we assume to be 2D.

Constraint generation:

$\max_{\pi} \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R_w(s_t) \mid \pi\right] = \max_{\pi} w^T \mu(\pi)$
Three broad categories of formalizations

- Max margin (Ratliff et al, 2006)
  - Feature boosting [Ratliff et al, 2007]
  - Hierarchical formulation [Kolter et al, 2008]

- Feature expectation matching (Abbeel + Ng, 2004)
  - Two player game formulation of feature matching (Syed + Schapire, 2008)
  - Max entropy formulation of feature matching (Ziebart et al, 2008)

- Interpret reward function as parameterization of a policy class. (Neu + Szepesvari, 2007; Ramachandran + Amir, 2007)

Feature expectation matching

- Inverse RL starting point: find a reward function such that the expert outperforms other policies

Let \( R(s) = w^\top \phi(s) \), where \( w \in \mathbb{R}^n \), and \( \phi : S \to \mathbb{R}^n \).

Find \( w^* \) such that \( w^* \mu(\pi^*) \geq w^* \mu(\pi) \quad \forall \pi \)

- Observation in Abbeel and Ng, 2004: for a policy \( \pi \) to be guaranteed to perform as well as the expert policy \( \pi^* \), it suffices that the feature expectations match:

\[
\|\mu(\pi) - \mu(\pi^*)\| \text{ small implies } \|w^* \mu(\pi^*) - w^* \mu(\pi)\| \text{ small}
\]

\( \rightarrow \) How to find such a policy \( \pi \) ?
Feature expectation matching

- If expert suboptimal:
  - *Abbeel and Ng, 2004*: resulting policy is a mixture of policies which have expert in their convex hull. In practice: pick the best one of this set and pick the corresponding reward function.
  
  - *Syed and Schapire, 2008*: recast the same problem in game theoretic form which, at cost of adding in some prior knowledge, results in having a unique solution for policy and reward function.
  
  - *Ziebart+al, 2008*: assume the expert stochastically chooses between paths where each path’s log probability is given by its expected sum of rewards.

Lecture outline

- Inverse RL intro
- Mathematical formulations for inverse RL
  - Max-margin
  - Feature matching
    - Reward function parameterizing the policy class
- Case studies
Reward function parameterizing the policy class

- Recall:
  \[ V^*(s; R) = R(s) + \gamma \max_a \sum_{s'} P(s'|s, a) V^*(s; R) \]
  \[ Q^*(s, a; R) = R(s) + \gamma \sum_{s'} P(s'|s, a) V^*(s; R) \]

- Let’s assume our expert acts according to:
  \[ \pi(a|s; R, \alpha) = \frac{1}{Z(s; R, \alpha)} \exp(\alpha Q^*(s, a; R)) \]

- Then for any \( R \) and \( \alpha \), we can evaluate the likelihood of seeing a set of state-action pairs as follows:
  \[ P((s_1, a_1)) \ldots P((s_m, a_m)) = \frac{1}{Z(s_1; R, \alpha)} \exp(\alpha Q^*(s_1, a_1; R)) \ldots \frac{1}{Z(s_m; R, \alpha)} \exp(\alpha Q^*(s_m, a_m; R)) \]

- Note: non-convex formulation --- due to non-linear equality constraint for \( V! \)
- Ramachandran and Amir, AAAI2007: MCMC method to sample from this distribution
- Neu and Szepesvari, UAI2007: gradient method to find local optimum of the likelihood

Lecture outline

- Inverse RL intro
- Mathematical formulations for inverse RL
- Case studies:
  - Highway driving,
  - Parking lot navigation,
  - Route inference,
  - Quadruped locomotion
Simulated highway driving

Abbeel and Ng, ICML 2004; Syed and Schapire, NIPS 2007

Highway driving

Teacher in Training World

Learned Policy in Testing World

- Input:
  - Dynamics model / Simulator \( P_s(s_{t+1} | s_t, a_t) \)
  - Teacher’s demonstration: 1 minute in “training world”
  - Note: \( R^* \) is unknown.
  - Reward features: 5 features corresponding to lanes/shoulders; 10 features corresponding to presence of other car in current lane at different distances

[Abbeel and Ng 2004]
More driving examples

In each video, the left sub-panel shows a demonstration of a different driving “style”, and the right sub-panel shows the behavior learned from watching the demonstration.

Parking lot navigation

- Reward function trades off:
  - Staying “on-road,”
  - Forward vs. reverse driving,
  - Amount of switching between forward and reverse,
  - Lane keeping,
  - On-road vs. off-road,
  - Curvature of paths.

[Abbeel and Ng 2004]

[Abbeel et al., IROS 08]
Experimental setup

- Demonstrate parking lot navigation on “train parking lots.”
- Run our apprenticeship learning algorithm to find the reward function.
- Receive “test parking lot” map + starting point and destination.
- Find the trajectory that maximizes the learned reward function for navigating the test parking lot.

Nice driving style
Sloppy driving-style

Only 35% of routes are "fastest" (Letchner, Krumm, & Horvitz 2006)
Data Collection

- Length
- Speed
- Road Type
- Lanes
- Accidents
- Construction
- Congestion
- Time of day

25 Taxi Drivers
Over 100,000 miles
Ziebart+al, 2007/8/9

Destination Prediction
Quadruped

- Reward function trades off 25 features.

Hierarchical max margin [Kolter, Abbeel & Ng, 2008]

Experimental setup

- Demonstrate path across the “training terrain”

- Run our apprenticeship learning algorithm to find the reward function

- Receive “testing terrain”---height map.

- Find the optimal policy with respect to the learned reward function for crossing the testing terrain.

Hierarchical max margin [Kolter, Abbeel & Ng, 2008]
Without learning

With learned reward function
Inverse RL history

- 1964, Kalman posed the inverse optimal control problem and solved it in the 1D input case
- 1994, Boyd+al.: a linear matrix inequality (LMI) characterization for the general linear quadratic setting
- 2000, Ng and Russell: first MDP formulation, reward function ambiguity pointed out and a few solutions suggested
- 2004, Abbeel and Ng: inverse RL for apprenticeship learning---reward feature matching
- 2006, Ratliff+al: max margin formulation
Inverse RL history

- 2007, Ratliff+al: max margin with boosting—enables large vocabulary of reward features
- 2007, Ramachandran and Amir, and Neu and Szepesvari: reward function as characterization of policy class
- 2008, Kolter, Abbeel and Ng: hierarchical max-margin
- 2008, Syed and Schapire: feature matching + game theoretic formulation
- 2008, Ziebart+al: feature matching + max entropy
- 2008, Abbeel+al: feature matching -- application to learning parking lot navigation style
- Active inverse RL? Inverse RL w.r.t. minmax control, partial observability, learning stage (rather than observing optimal policy), … ?