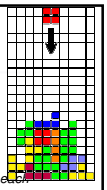
CS 287: Advanced Robotics
Fall 2009

Lecture 16: imitation learning

Pieter Abbeel
UC Berkeley EECS

Behavioral cloning example



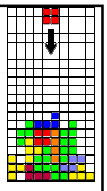
$V(s) = \sum_{i=1}^{22} \theta_i \phi_i(s)$

- 22 features aka basis functions ϕ_i
  - Ten basis functions, 0, . . . , *9, mapping the state to the height h[k] of each of the ten* columns.
  - Nine basis functions, 10, . . . , *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.

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

Behavioral cloning example

Behavioral cloning example



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

Behavioral cloning example

Behavioral cloning example

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

Max-margin formulation

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

$$\text{subject to} \quad \forall i, \forall j : \theta^\top \phi(s_+^{(i)}) \geq \theta^\top \phi(s_{j-}^{(i)}) + 1 - \xi_{i,j}$$

Probabilistic/Logistic formulation

Assumes experts choose for result $s^{(i)}$ with probability $\frac{\exp(\theta^\top \phi(s_+^{(i)}))}{\exp(\theta^\top \phi(s_+^{(i)})) + \sum_{j-} \exp(\theta^\top \phi(s_{j-}^{(i)}))}$.
Hence the maximum likelihood estimate is given by:

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

Lecture outline

- Inverse RL intro
- *Mathematical formulations for inverse RL*
- Case studies

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

Three broad categories of formalizations

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

Problem setup

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

Basic principle

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

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

- 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 π^* --- 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 $w \in \Re^n$, and $\phi : S \to \Re^n$.

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

Feature based reward function

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

$$\begin{aligned} \mathrm{E}[\sum_{t=0}^{\infty} \gamma^t R(s_t)|\pi] &= \mathrm{E}[\sum_{t=0}^{\infty} \gamma^t w^\top \phi(s_t)|\pi] \\ &= w^\top \mathrm{E}[\sum_{t=0}^{\infty} \gamma^t \phi(s_t)|\pi] \\ &= w^\top \mu(\pi) \end{aligned}$$

Expected cumulative discounted sum of feature values or "feature expectations"

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

$$\text{Find } w^* \text{ such that } w^{*\top}\mu(\pi^*) \geq w^{*\top}\mu(\pi) \quad \forall \pi$$

Feature based reward function

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

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

- 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 π^*
  - Size of the state space

Recap of challenges

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

- Challenges:
  - Assumes we know the entire expert policy π^* → 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 $w^{*\top}\mu(\pi^*) \geq w^{*\top}\mu(\pi) \quad \forall \pi$
- Standard max margin:
- "Structured prediction" max margin:

Ambiguity

- Standard max margin:

$$\begin{aligned} &\min_w \|w\|_2^2 \\ &\text{s.t. } w^\top \mu(\pi^*) \geq w^\top \mu(\pi) + 1 \quad \forall \pi \end{aligned}$$

- "Structured prediction" max margin:

$$\begin{aligned} &\min_w \|w\|_2^2 \\ &\text{s.t. } w^\top \mu(\pi^*) \geq w^\top \mu(\pi) + m(\pi^*, \pi) \quad \forall \pi \end{aligned}$$

- Justification: margin should be larger for policies that are very different from π^*.
- Example: m(π, π^*) = number of states in which π^* was observed and in which π and π^* disagree

Expert suboptimality

- Structured prediction max margin:

$$\min_w \|w\|_2^2$$
$$\text{s.t.} \quad w^\top \mu(\pi^*) \geq w^\top \mu(\pi) + m(\pi^*, \pi) \quad \forall \pi$$

Constraint generation

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

- Solve

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

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

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

  = 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.

Expert suboptimality

- Structured prediction max margin with slack variables:

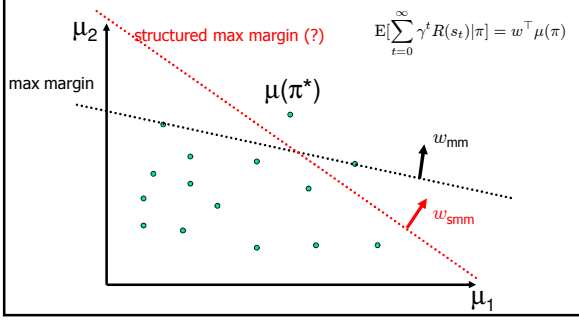
$$\min_{w,\xi} \|w\|_2^2 + C\xi$$
$$\text{s.t.} \quad w^\top \mu(\pi^*) \geq w^\top \mu(\pi) + m(\pi^*, \pi) - \xi \quad \forall \pi$$

- Can be generalized to multiple MDPs (could also be same MDP with different initial state)

$$\min_{w,\xi^{(i)}} \|w\|_2^2 + C \sum_i \xi^{(i)}$$
$$\text{s.t.} \quad w^\top \mu(\pi^{(i)*}) \geq w^\top \mu(\pi^{(i)}) + m(\pi^{(i)*}, \pi^{(i)}) - \xi^{(i)} \quad \forall i, \pi^{(i)}$$

Visualization in feature expectation space

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



Complete max-margin formulation

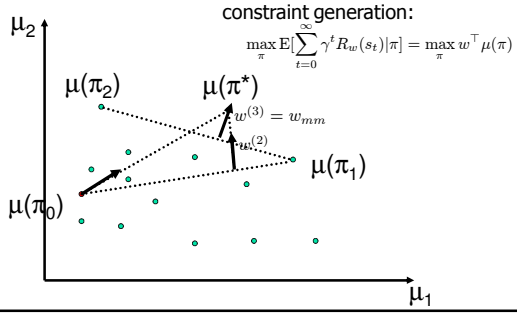
$$\min_w \|w\|_2^2 + C \sum_i \xi^{(i)}$$
$$\text{s.t.} \quad w^\top \mu(\pi^{(i)*}) \geq w^\top \mu(\pi^{(i)}) + m(\pi^{(i)*}, \pi^{(i)}) - \xi^{(i)} \quad \forall i, \pi^{(i)}$$

[Ratliff, Zinkevich and Bagnell, 2006]

- Resolved: access to π^*, ambiguity, expert suboptimality
- *One challenge remains: very large number of constraints*
  - *Ratliff+al use subgradient methods.*
  - *In this lecture: constraint generation*

Constraint generation

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



Three broad categories of formalizations

- Max margin (Ratliff+al, 2006)
  - Feature boosting [Ratliff+al, 2007]
  - Hierarchical formulation [Kolter+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+al,2008)*
- Interpret reward function as parameterization of a policy class. (Neu+Szepesvari, 2007; Ramachandran+Amir, 2007)

Lecture outline

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

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 $w \in \Re^n$, and $\phi : S \rightarrow \Re^n$.

Find w^* such that $w^{*\top} \mu(\pi^*) \geq w^{*\top} \mu(\pi) \quad \forall \pi$

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

$$\|\mu(\pi) - \mu(\pi^*)\| \text{ small implies } \|w^{*\top}\mu(\pi^*) - w^{*\top}\mu(\pi)\| \text{ small}$$

→ How to find such a policy π ?

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 α, 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

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
- *Case studies:*
  - *Highway driving,*
  - *Parking lot navigation,*
  - *Route inference,*
  - *Quadruped locomotion*

Simulated highway driving

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

Parking lot navigation

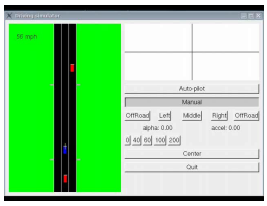
- Reward function trades off:
  - Staying "on-road,"
  - Forward vs. reverse driving,
  - Amount of switching between forward and reverse,
  - Lane keeping,
  - On-road vs. off-road,
  - Curvature of paths.

[Abbeel et al., IROS 08]

Highway driving

[Abbeel and Ng 2004]

Teacher in Training World

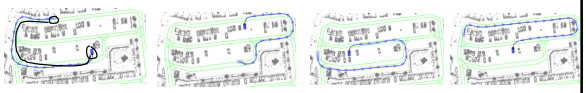


Learned Policy in Testing World

- Input:
  - Dynamics model / Simulator $P_{sa}(s_{t+1} \mid s_t, a_t)$
  - 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

Experimental setup

- Demonstrate parking lot navigation on "train parking lots."

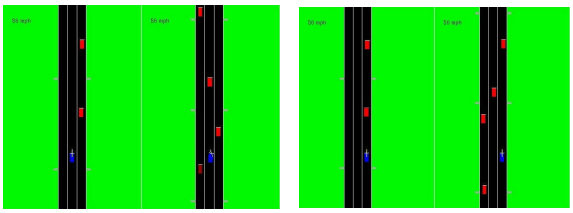


- Run our apprenticeship learning algorithm to find the reward function.
- Receive "test parking lot" map + starting point and destination.
- Find the trajectory that maximizes the *learned reward function* for navigating the test parking lot.

More driving examples

[Abbeel and Ng 2004]

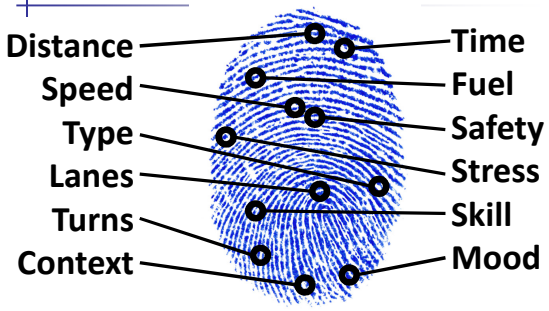
Driving demonstration | Learned behavior | Driving demonstration | Learned behavior



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.

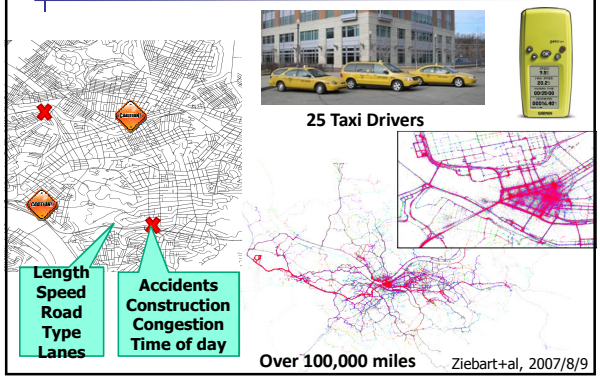
Nice driving style

Sloppy driving-style



Ziebart+al, 2007/8/9

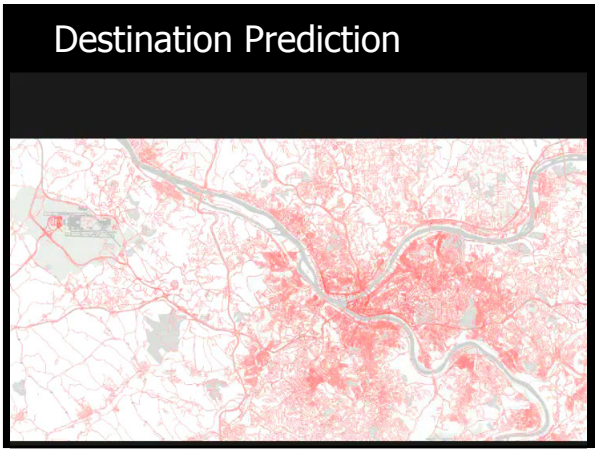
Data Collection



Ziebart+al, 2007/8/9

Ziebart+al, 2007/8/9

Destination Prediction



Quadruped

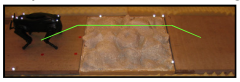
- Reward function trades off 25 features.

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

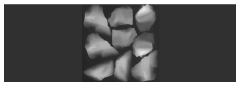
With learned reward function

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

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), … ?