Inverse Reinforcement Learning in Animal Behavior Characterization

Joschka Boedecker **Hao Zhu**

Department of Computer Science
University of Freiburg

September 25, 2024

About this talk

- mathematical optimization and inverse reinforcement learning, sloppy math
- IRL algorithms for behavior characterization, done by others and us
- examples and opinions (some controversial)
- no theoretical/technical details (convergence analysis, solver implementation, *etc.*)
- sadly, no fun videos

Outline

Optimization problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & h_i(x) = 0, \quad i = 1, \ldots, p \end{array}$$

- $x \in \mathbf{R}^n$ is (vector) *variable* to be chosen
- f_0 is the *objective function*, to be minimized
- $f_1, \ldots, f_m$ are the *inequality constraint functions*
- $h_1, \ldots, h_p$ are the *equality constraint functions*

- variations: maximize objective, multiple objectives, ...
- applications: trades in a portfolio, engineering design, airplane control surface deflections, resource allocation, ...

Finding good models

- x represents the *parameters* in a model
- constraints impose requirements on model parameters
- objective $f_0(x)$ is the prediction error on some observed data (and possibly a term that penalized model complexity)

example: standard support vector classifier

$$\begin{array}{ll} \text{minimize} & \|a\|_2 + \gamma(\mathbf{1}^T u + \mathbf{1}^T v) \\ \text{subject to} & a^T x_i + b \geq 1 - u_i, \quad i = 1, \ldots, N \\ & a^T y_i + b \leq -(1 - v_i), \quad i = 1, \ldots, M \\ & u \succeq 0, \quad v \succeq 0 \end{array}$$

- a is the normal of the support hyperplane
- u, v are measure of how much the linear discrimination constraints are violated
- produces point on trade-off curve between inverse of margin $2/\|a\|_2$ and classification error, measured by total slack $\mathbf{1}^T u + \mathbf{1}^T v$

Matrix norm minimization

$$\text{minimize} \quad \|A(x)\|_2 = \left(\lambda_{\max}(A(x)^T A(x))\right)^{1/2}$$

where $A(x) = A_0 + x_1 A_1 + \cdots + x_n A_n$ (with given $A_i \in \mathbf{R}^{p\times q}$)

equivalent semi-definite program

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & \begin{bmatrix} tI & A(x) \\ A(x)^T & tI \end{bmatrix} \succeq 0 \end{array}$$

- variables $x \in \mathbf{R}^n$, $t \in \mathbf{R}$
- can be solved **very fast** using interior-point methods
  (almost always in 10 to 100 iterations with each in polynomial time)

Inversion

- x is something we want to estimate/reconstruct, given some measurement y
- constraints come from prior knowledge about x
- objective $f_0(x)$ measures deviation between predicted and actual measurements

Outline

Markov decision processes (MDPs)

a finite MDP is a 5-tuple

$$(\mathcal{S}, \mathcal{A}, P, r, \gamma)$$

where

- $\mathcal{S}$ is the *state space*
- $\mathcal{A}$ is the *action space*
- $P\colon \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the *state transition function*
- $r\colon \mathcal{S} \times \mathcal{A} \to \mathbf{R}$ is the *reward function*
- $\gamma \in [0, 1]$ is the *discount factor*

- $\pi\colon \mathcal{S} \times \mathcal{A} \to [0, 1]$ is agent's *policy*

Inverse reinforcement learning (IRL)

- is also called (modern) *inverse optimal control*
- consists in estimating a reward function r that best explains some expert's behavior, given the set of expert demonstrations $\mathcal{D}$
- the naive formulation is ill-posed
  i.e., many reward functions would explain the behavior
  e.g., (informal) any $r(s, a) = c$ for all $s \in \mathcal{S}$, $a \in \mathcal{A}$ with $c \in \mathbf{R}$ is optimal

Maximum (causal) entropy IRL

(Ziebart *et al.* [ZMB+08])

maximize (over π) $\quad \mathbf{E}_{\xi\sim\pi,P}\left[-\sum_{t=0}^{n}\gamma^t \log \pi(s_t,a_t)\right]$

subject to $\quad \mathbf{E}_{\xi\sim\pi,P}\left[\sum_{t=0}^{n}\gamma^t r(s_t,a_t)\right] = \mathbf{E}_{\xi\sim\mathcal{D}}\left[\sum_{t=0}^{n}\gamma^t r(s_t,a_t)\right]$

- $\xi = \{(s_0,a_0),\ldots,(s_n,a_n)\}$ — trajectory
- maximizing causal entropy of policy under reward matching constraint

- the most widely used type of IRL formulation
- variants: guided cost learning, adversarial IRL, ...
- hard (long time, *etc.*) to solve

Inverse Q-learning

(Kalweit *et al.* [KHWB20])

maximize (over r)	$\mathbf{E}_{\xi \sim \mathcal{D}} \left[\log \mathbf{P} \left(\xi \mid \pi_r \right) \right]$
subject to	$\pi_r(s,a) = \exp\left(Q(s,a) - \log \sum \exp Q(s,\cdot)\right),$
	$Q(s,a) = r(s,a) + \gamma \sum_{s' \in \mathcal{S}} P(s,a,s') \max_{a' \in \mathcal{A}} Q(s',a'),$
	for all $s \in \mathcal{S}$, $a \in \mathcal{A}$

- maximum-likelihood estimation with Boltzmann policy constraint
- the transition function P need not be known, closed-form solution if known
- faster than maximum entropy IRL

Outline

Dealing with inconsistent demonstrations

stationary reward function does not work well on demonstrations from

- multiple experts with individual preferences
- suboptimal experts
- environments consists of multiple (sequential) contradicting subtasks (*e.g.*, the infamous Atari game *Montezuma's Revenge*)
- animals (including human in some environments), whose goals can evolve over time due to, *e.g.*, fatigue, satiation, and curiosity

Dynamical inverse reinforcement learning (DIRL)

(Ashwood *et al.* [AJP22])

extend maximum entropy IRL on a continuous, smoothly varying reward function:

$$r_t(s) = \sum_{k=1}^{K} \alpha_{k,t} u_k(s)$$

- $u_k \in \mathbf{R}^{|\mathcal{S}|}$ — the kth reward function
- $\alpha_{k,t} \in \mathbf{R}$ — mixing weight, where $\alpha_{k,t} = \alpha_{k,t-1} + \epsilon_k$ with $\epsilon_k \sim \mathcal{N}(0, \sigma_k^2)$

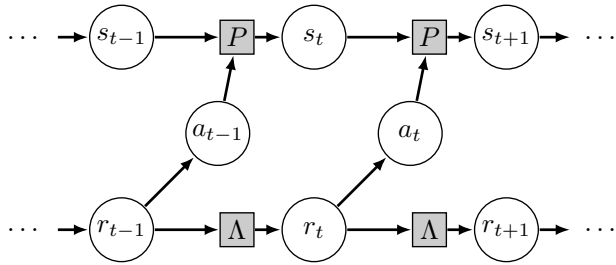
- achieved SOTA in animal behavior prediction

Outline

Hierarchical inverse Q-learning (HIQL)

(Zhu *et al.* [ZDK+24])

- each expert demonstration is generated under one of the reward functions in the set $\mathcal{R} = \{r_1, \ldots, r_K\}$
- the probability that one demonstration is generated under $r \in \mathcal{R}$ is controlled by a Markov chain with initial state distribution Π and transition matrix Λ



Optimization problems

iteratively solving the sequence of problems:

$$\begin{array}{ll} \text{maximize (over } \Pi^+) & \mathbf{E}_{\xi\sim\mathcal{D}}\left[\sum_{i=1}^K \mathbf{P}(z_0 = i \mid \xi, \Theta) \log \Pi_i^+\right] \\ \text{subject to} & \Pi^+ \succeq 0,\ \mathbf{1}^T\Pi^+ = 1 \end{array}$$

$$\begin{array}{ll} \text{maximize (over } \Lambda^+) & \mathbf{E}_{\xi\sim\mathcal{D}}\left[\sum_{i=1}^K \sum_{j=1}^K \sum_{t=1}^n \mathbf{P}(z_{t-1} = i, z_t = j \mid \xi, \Theta) \log \Lambda_{ij}^+\right] \\ \text{subject to} & \Lambda_{ij}^+ \geq 0, \quad i,j = 1,\ldots,K, \quad \Lambda\mathbf{1} = \mathbf{1} \end{array}$$

$$\begin{array}{ll} \text{maximize (over } r_i^+) & \mathbf{E}_{\xi\sim\mathcal{D}}\left[\sum_{t=0}^n \mathbf{P}(z_t = i \mid \xi, \Theta) \log \pi_{r_i^+}(s_t, a_t)\right] \\ \text{subject to} & \pi_{r_i^+}(s,a) = \exp\left(Q(s,a) - \log \sum \exp Q(s,\cdot)\right), \\ & Q(s,a) = r_i^+(s,a) + \gamma \sum_{s'\in\mathcal{S}} P(s,a,s') \max_{a'\in\mathcal{A}} Q(s',a'), \\ & \quad \text{for all } s \in \mathcal{S},\ a \in \mathcal{A} \end{array}$$

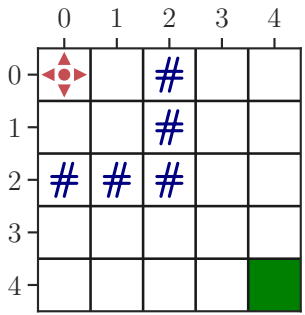
until $\Theta = \{r_1, \ldots, r_K, \Pi, \Lambda\}$ converges

- analytical solution exists for Π^+ and Λ^+
- adapt inverse Q-learning for estimating $r_1^+, \ldots, r_K^+$

Outline

Gridworld

environment



- $\mathcal{A} = \{$*left*, *right*, *up*, *down*, *stay*$\}$
- 10% probability to random state

expert behavior

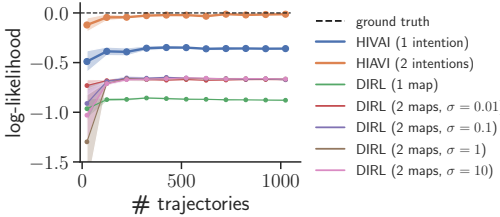
- π^{goal}: move towards $(4, 4)$
- π^{abandon}: move towards $(0, 0)$

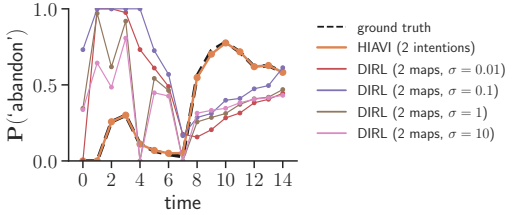
---

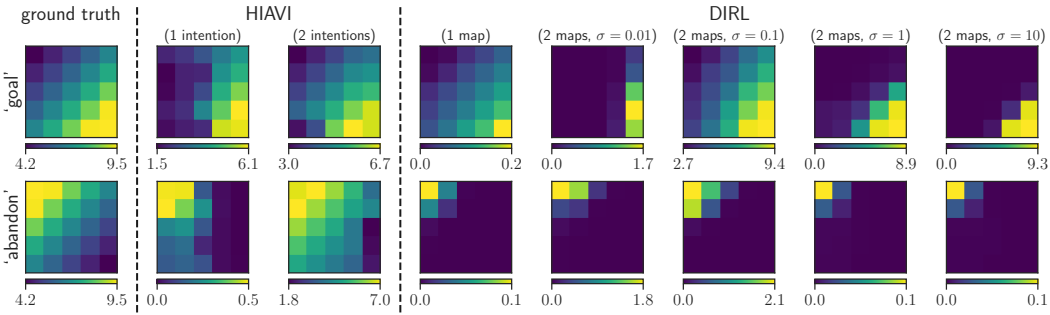
initialize $s := (0, 0)$, $\pi := \pi^{\text{goal}}$, $t := 0$.
repeat
  $a \sim \pi$.
  $s \sim P(s, a, \cdot)$.
  if s has barrier '#' **then**
    Switch to another policy (30%).
  else if $t = 8$ **then**
    $\pi := \pi^{\text{abandon}}$ (50%).
  end if
  $t := t + 1$.
until $(0, 0)$ or $(4, 4)$ is reached.

---

results

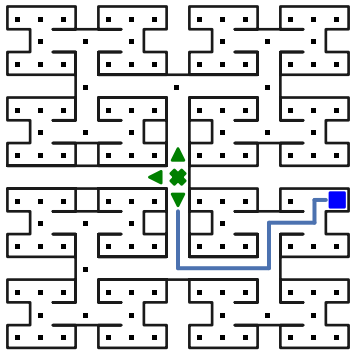

log-likelihood
0.0
−0.5
−1.0
−1.5
0
500
1000
trajectories
ground truth
HIVAI (1 intention)
HIVAI (2 intentions)
DIRL (1 map)
DIRL (2 maps, $\sigma = 0.01$)
DIRL (2 maps, $\sigma = 0.1$)
DIRL (2 maps, $\sigma = 1$)
DIRL (2 maps, $\sigma = 10$)


P('abandon')
1.0
0.5
0.0
0
2
4
6
8
10
12
14
time
ground truth
HIAVI (2 intentions)
DIRL (2 maps, $\sigma = 0.01$)
DIRL (2 maps, $\sigma = 0.1$)
DIRL (2 maps, $\sigma = 1$)
DIRL (2 maps, $\sigma = 10$)


ground truth
HIAVI
(1 intention)
(2 intentions)
DIRL
(1 map)
(2 maps, $\sigma = 0.01$)
(2 maps, $\sigma = 0.1$)
(2 maps, $\sigma = 1$)
(2 maps, $\sigma = 10$)
'goal'
'abandon'
4.2
9.5
1.5
6.1
3.0
6.7
0.0
0.2
0.0
1.7
2.7
9.4
0.0
8.9
0.0
9.3
4.2
9.5
0.0
0.5
1.8
7.0
0.0
0.1
0.0
1.8
0.0
2.1
0.0
0.1
0.0
0.1

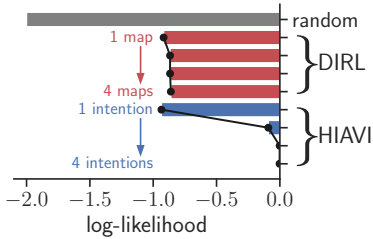
Mice labyrinth navigation

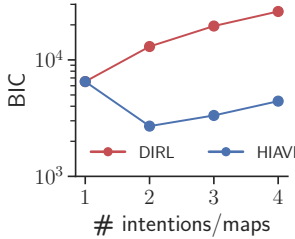
environment

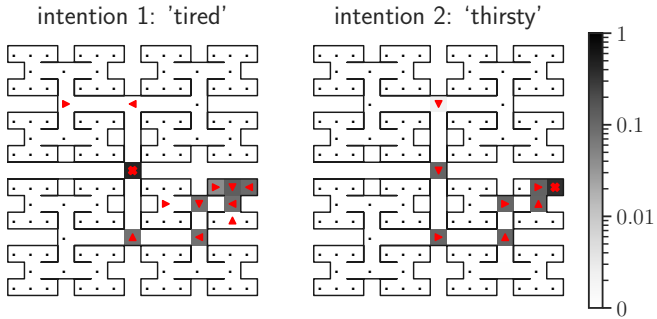


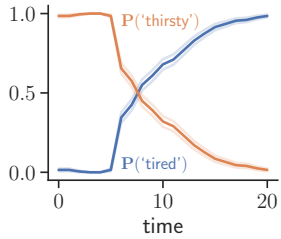
- action space: $\mathcal{A} = \{\textit{left}, \textit{right}, \textit{reverse}, \textit{stay}\}$
- subjects: water-restricted & -unrestricted mice

water-restricted animals

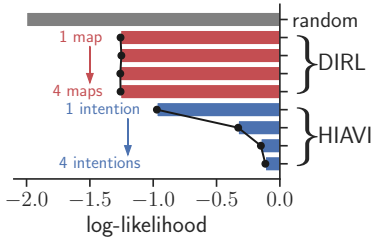


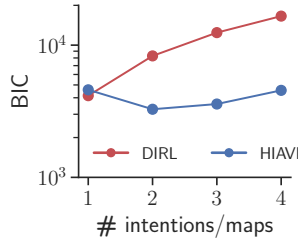


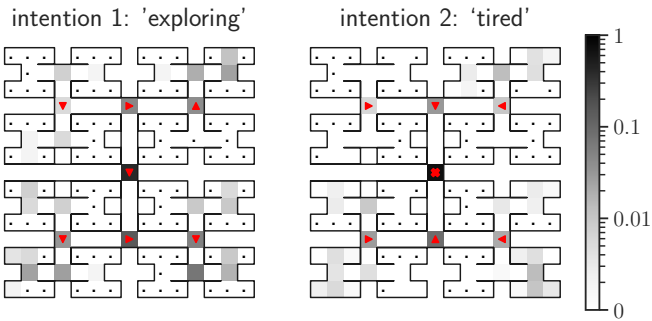


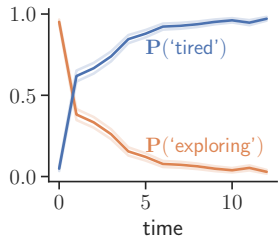


water-unrestricted animals









Outline

Conclusion (non-controversial)

HIQL

- outperforms the SOTA in animal behavior prediction
- provides interpretable behavior representations

Conclusion (controversial)

- the intention transition dynamics under natural decision-making processes aligns better with a discrete Markov process (step-like function), compared to a continuous random walk process (smoothly varying function)

- *tuned mathematical optimization inverse models can help with animal behavior characterization under complex environments*

Challenges

- IRL problems with **nonconvex** formulation is very hard to solve[1]
  - local optimal
  - non-polynomial time complexity
  - sensitive to initial guess, *etc.*
- scale up to high-dimensional (*e.g.*, continuous space) environments

[1] normally; depends on the definition of "solve"

Reference

[AJP22] Zoe Ashwood, Aditi Jha, and Jonathan W Pillow.
Dynamic inverse reinforcement learning for characterizing animal behavior.
In *NeurIPS*, volume 35, pages 29663–29676, 2022.

[KHWB20] Gabriel Kalweit, Maria Huegle, Moritz Werling, and Joschka Boedecker.
Deep inverse Q-learning with constraints.
In *NeurIPS*, volume 33, pages 14291–14302, 2020.

[ZDK+24] Hao Zhu, Brice De La Crompe, Gabriel Kalweit, Artur Schneider, Maria Kalweit, Ilka Diester, and Joschka Boedecker.
Multi-intention inverse Q-learning for interpretable behavior representation.
Transactions on Machine Learning Research, 2024.

[ZMB+08] Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, Anind K Dey, et al.
Maximum entropy inverse reinforcement learning.
In *AAAI*, volume 8, pages 1433–1438. Chicago, IL, USA, 2008.

Non-Markovian reward transition

- each expert demonstration is generated under one of the reward functions in the set $\mathcal{R} = \{r_1, \ldots, r_K\}$
- for each trajectory $\xi = \{(s_1, a_1), \ldots, (s_n, a_n)\}$, there is a corresponding sequence of observations $\psi = \{\varphi_1, \ldots, \varphi_n\}$, with $\varphi \in \mathbf{R}^m$
- the probability that one demonstration is generated under $r \in \mathcal{R}$ is controlled by some vector-valued function $f_\theta \colon \mathbf{R}^m \to \Delta^{|\mathcal{R}|}$