Interactive Machine Learning: From Classifiers to Robotics

Brad Hayes

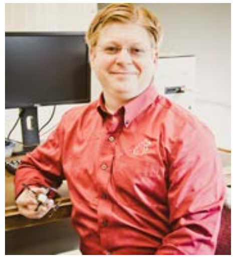
University of Colorado Boulder

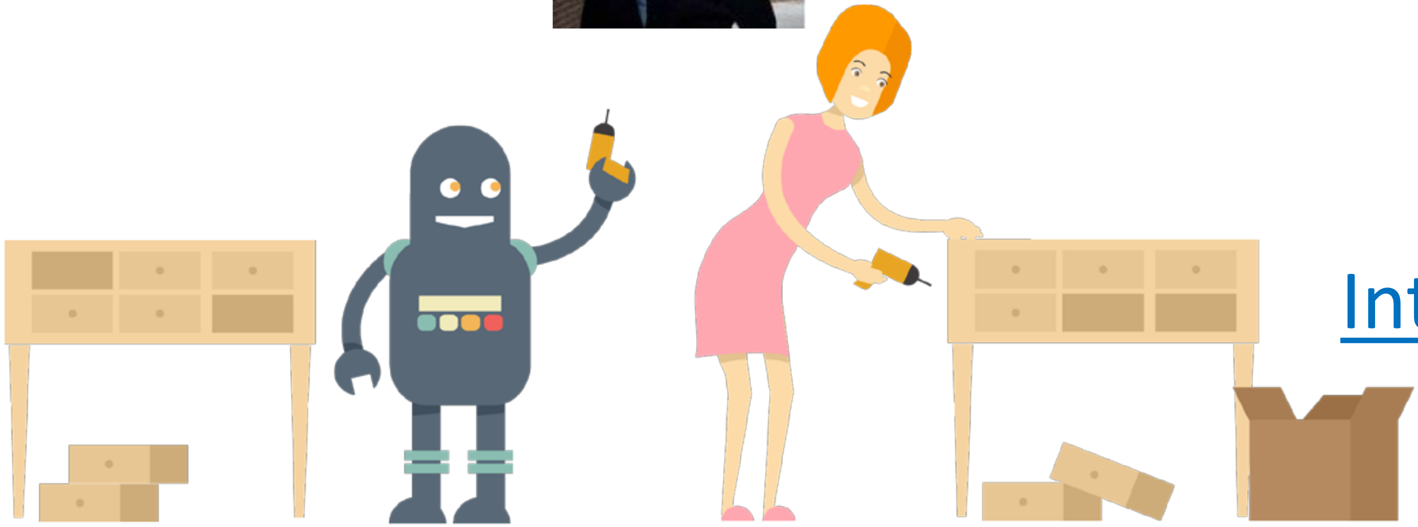
Ece Kamar

Microsoft Research

Matthew E. Taylor

Washington State University





InteractiveML.net

Training and Learning in Sequential Tasks

Sequential Tasks

- Versus pure Learning from Demonstration, we seek to:
  - Minimize uncertainty
  - Maximize smoothness
- Sequential Task
  - Implicitly or Explicitly looks ahead
  - Goal 1: Do what human wants you to do
  - Goal 2: Outperform human and outperform autonomous learning

Section Outline

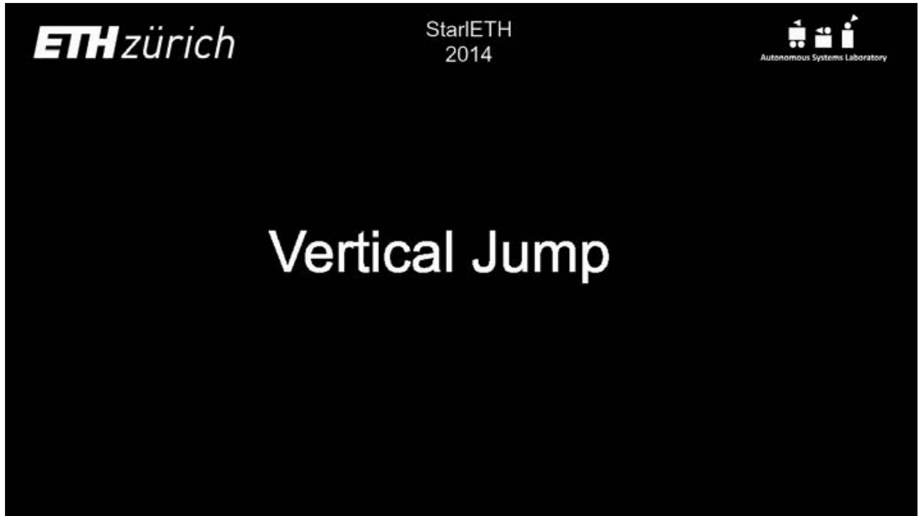
- Autonomous Learning: RL
- Demonstration + RL
  - action selection
  - shaping reward
  - IRL: shaping reward
- Learning from human feedback
  - Treat as environment reward
  - Treat as return
  - Return + RL
  - Treat as categorical feedback regrading policy

Reinforcement Learning (RL) Goals

- TD-Gammon beat Professionals: Tesauro, 1995
- Aibo Learned More Effective Gait: Kohl & Stone, 2004
- AlphaGo achieved Super-human performance: Silver et al., 2016

Learning autonomously is often better than hand-coding!

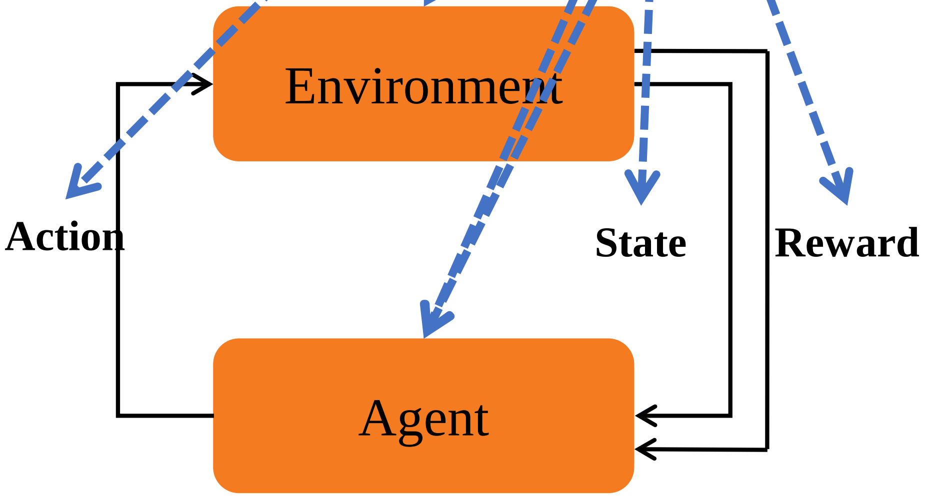
But not always!



RL Setting

Markov Decision Process (MDP)

- S: set of states in the world
- A: set of actions an agent can perform
- T: $S \times A \rightarrow S$ (transition function)
- R: $S \rightarrow \Re$ (environmental reward)
- π: $S \rightarrow A$ (policy)
- Q: $S \times A \rightarrow \Re$ (action-value function)



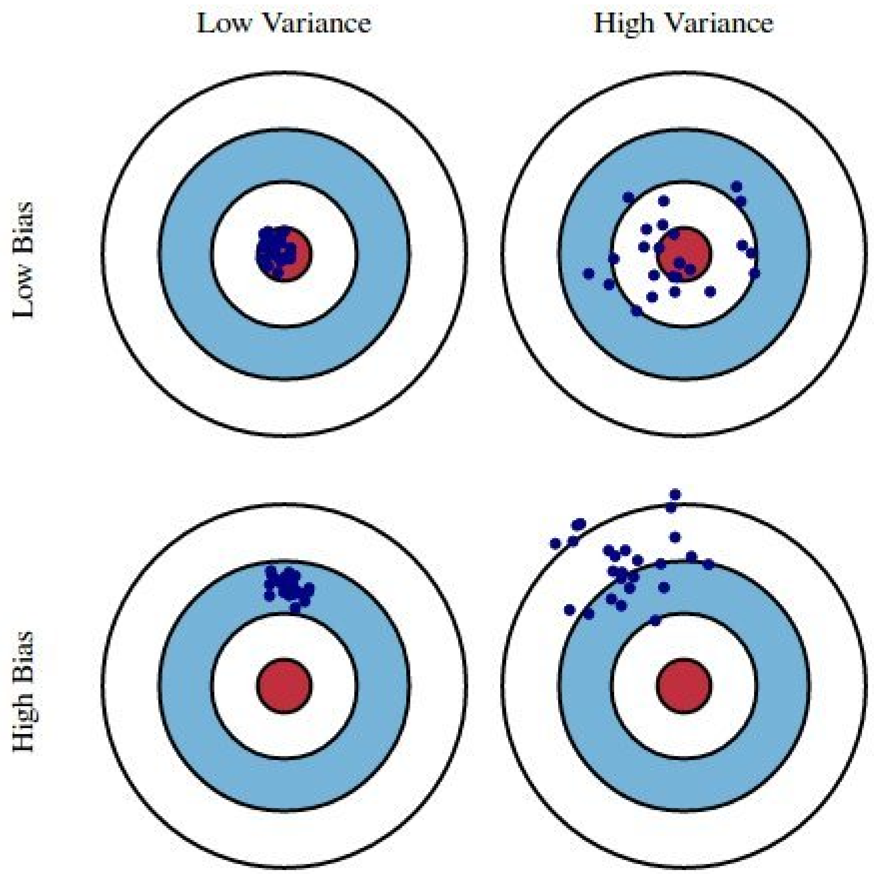
RL References

- Sutton & Barto, “Reinforcement Learning: An Introduction”
  https://webdocs.cs.ualberta.ca/~sutton/book/ebook/the-book.html
- Littman & Isbell Udacity course, “Reinforcement Learning”
  https://classroom.udacity.com/courses/ud600/
- Szepesvári, “Algorithms for Reinforcement Learning”
  https://sites.ualberta.ca/~szepesva/RLBook.html
- (Many others too)

RL & Speed

- Need data to learn. Can be equivalent to time
- Often start with random bias
- RL is worst at beginning (by definition?)
- Many techniques to achieve better Bias
  - Transfer Learning
  - Constrained action/state space
  - Hand-coded generalization

- Today: Bias from a human



Section Outline

- Autonomous Learning: RL
- Demonstration + RL
  - action selection
  - shaping reward
  - IRL: shaping reward
- Learning from human feedback
  - Treat as environment reward
  - Treat as return
  - Return + RL
  - Treat as categorical feedback regrading policy

HAT: Human-Agent Transfer

1. Observe Human Demonstration (or suboptimal controller)
2. Summarize Policy
3. Bootstrap Autonomous Learning with summarized policy

`Hold Ball, Pass`$_1$`, Pass`$_2$

- IF dist(K_1,T_1) > 4
  → Hold Ball
- ELSEIF ang(K_2,K_1,T_1) > 45
  → Pass_1
- ELSEIF …

"Integrating Reinforcement Learning with Human Demonstrations of Varying Ability". Taylor, Suay, & Chernova, 2011

HAT: Human-Agent Transfer

1. Human Demonstration
2. Summarize Policy
3. Autonomous Learning

P(Execute) =1:
tries to mimic human

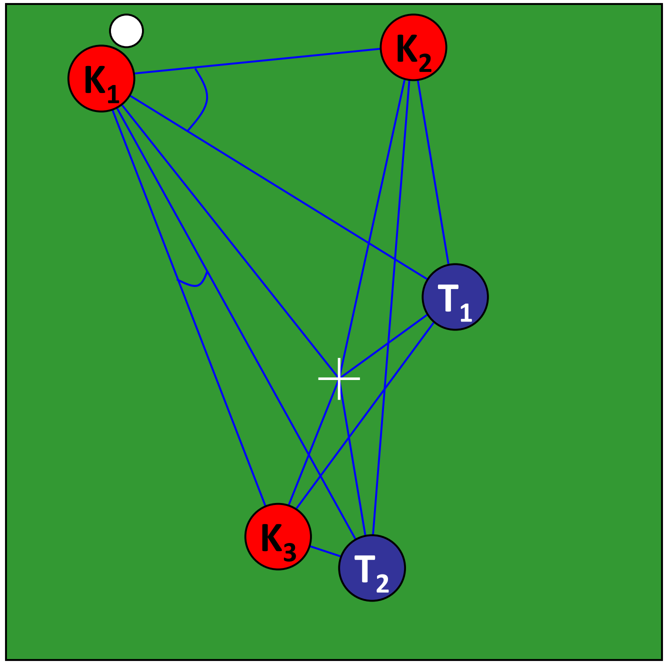
In state s, evaluate agent's 3 actions

$$Q(s, a_1) = 5$$
$$Q(s, a_2) = 3$$
$$Q(s, a_3) = 4$$

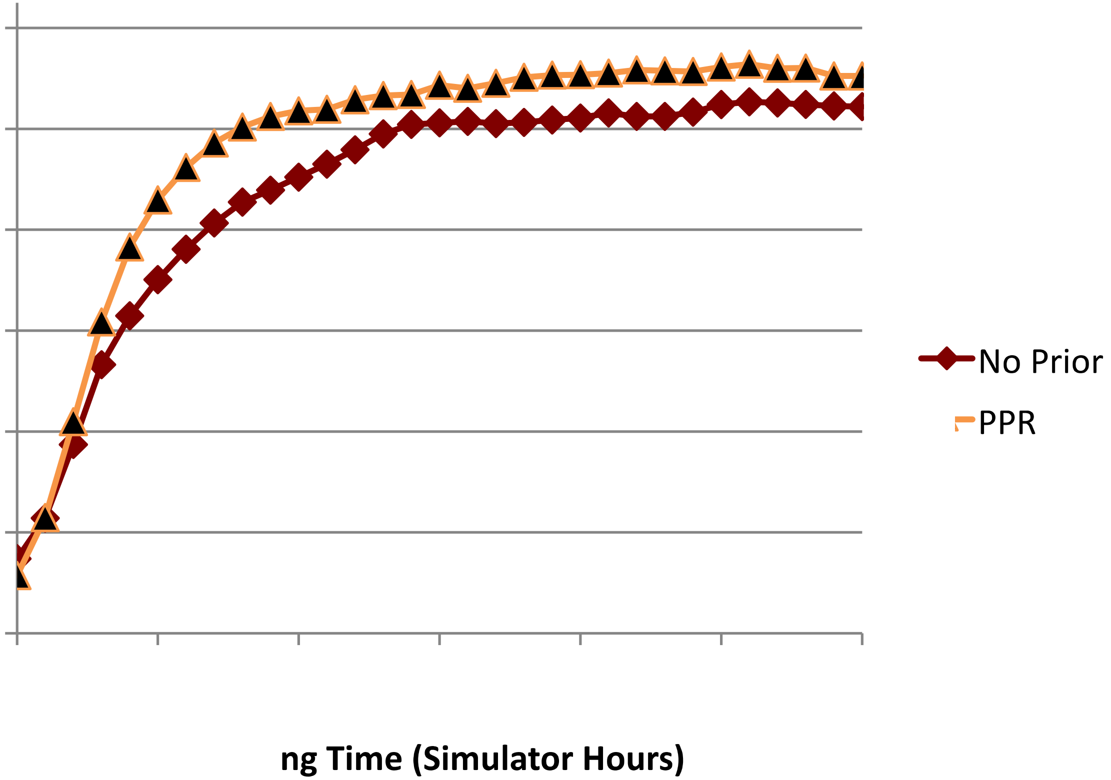
And evaluate action suggested by decision list

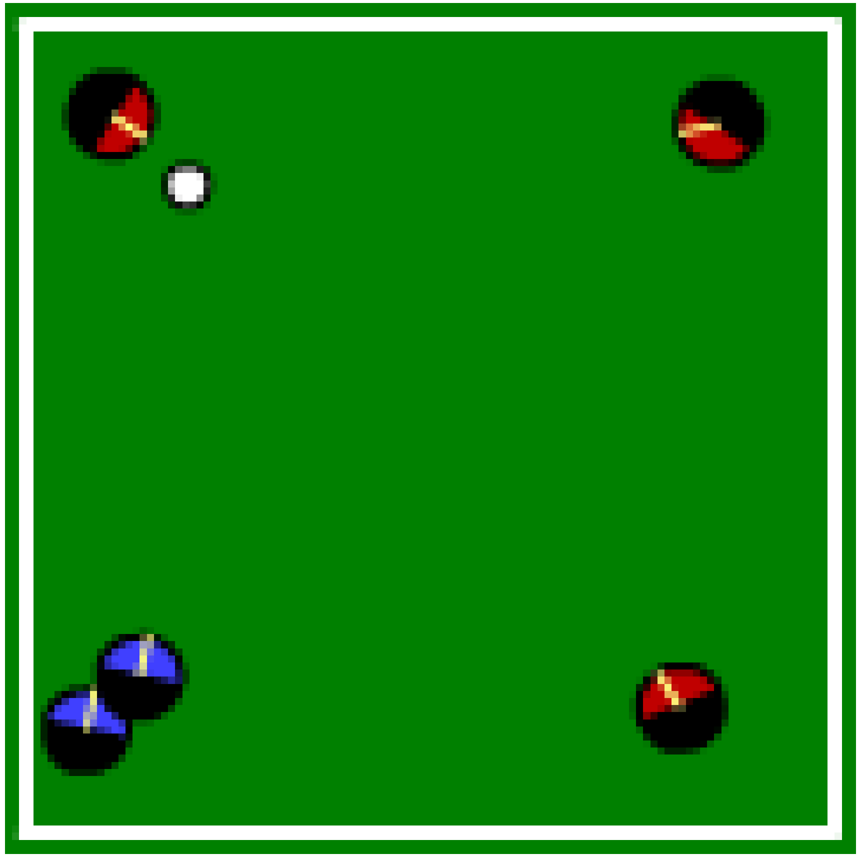
$$D_{target}(s) = a_3$$

- P(Execute): Take D(s) action
- P(Explore): Take random action
- P(Exploit): Take action w/ max Q



HAT: Human-Agent Transfer Results





Improvements with only ~3 minutes of human time

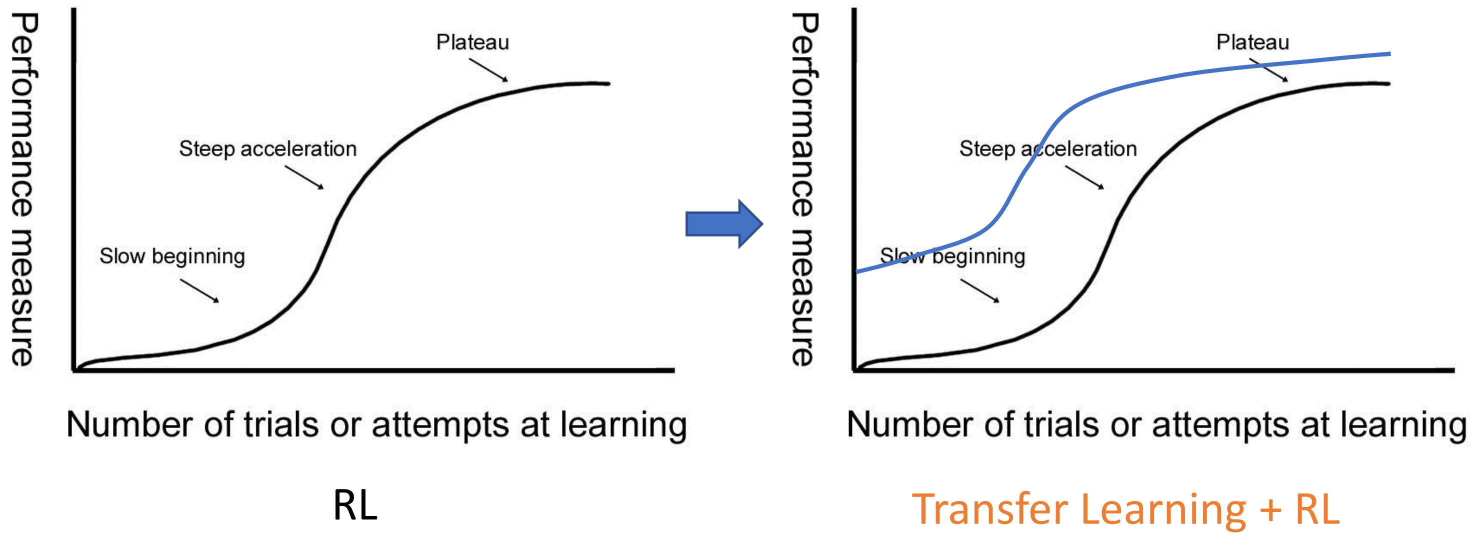
Example Policy

HAT: Human-Agent Transfer

Initiation: Student
Modality: Trajectories
Live/Offline: Offline
Present/Remote: N/A
Expertise: Any
Investment: N/A
Learning Paradigm: RL
Data Sources: Human provided + environment provided
Individual/Team Goal: Learner acts alone
Training/Testing: Tested on training task

Confidence HAT

- Goal: Improve Reinforcement Learning with Confidence-Based Demonstrations



“Improving Reinforcement Learning with Confidence-Based Demonstrations”. Wang & Taylor, IJCAI 2017

Confidence HAT

- Source demonstration quality?
- Source demonstration consistency?
- Summarization quality?
- Task coverage?

“Improving Reinforcement Learning with Confidence-Based Demonstrations”. Wang & Taylor, 2017

Confidence HAT

- 3-step method:

- Uncertainty measurement of demonstration
  - Summarize demonstration data into confidence-based models
  - Provide action suggestions along with confidence: let target agent decide (threshold)
  - Integrate with RL, help improve initial learning and overall performance

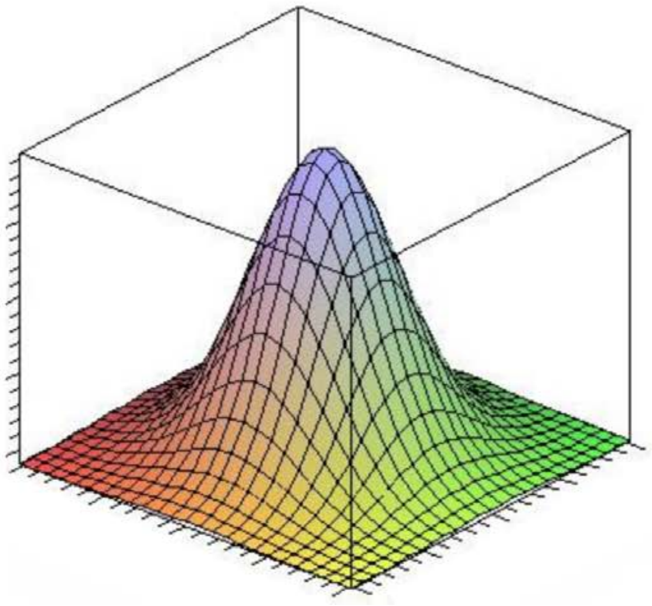
“Improving Reinforcement Learning with Confidence-Based Demonstrations”. Wang & Taylor, 2017

Where does the confidence model come from?

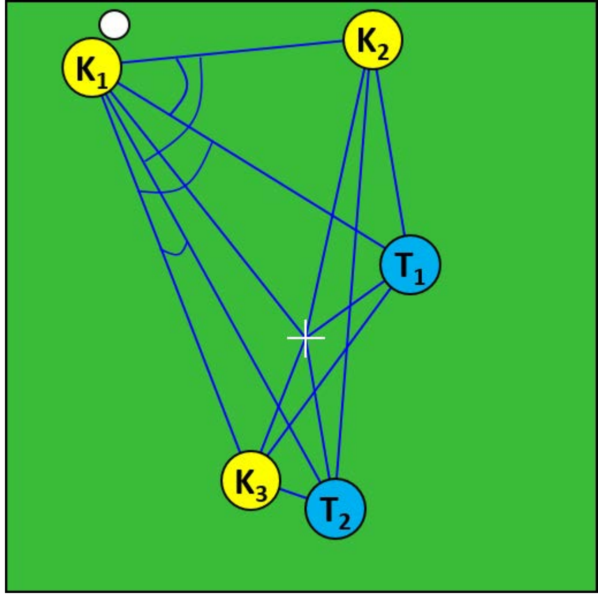
Summarize the prior knowledge into Gaussian Process model

$$P(\omega_i|x) = \frac{1}{\sqrt{2\pi|\Sigma_i|}} \exp\{-\frac{1}{2}(x-\mu_i)^T \Sigma_i^{-1}(x-\mu_i)\}$$

- Confidence Function



Keepaway Domain



Demonstration:

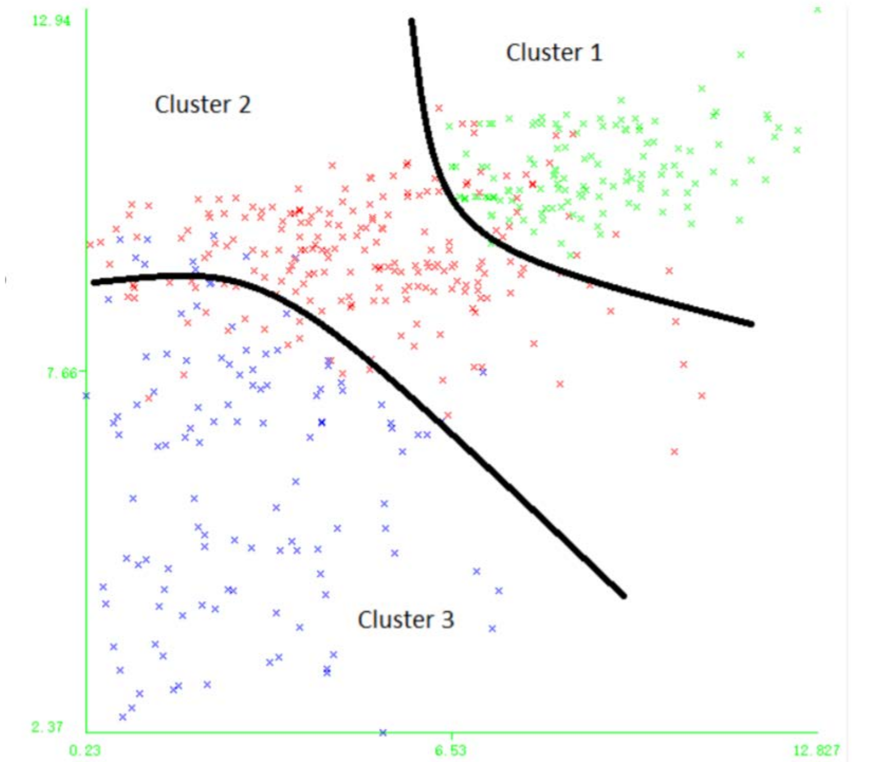
- State-action pairs of 20 episodes

GPHAT:

- Cluster active data (Pass1 & Pass2) into smaller groups.
- Train Gaussian classifiers upon smaller clusters.
- Set a threshold. Follow GPHAT's suggested action with confidence higher than that. (e.g. 0.9 is a reasonable value).

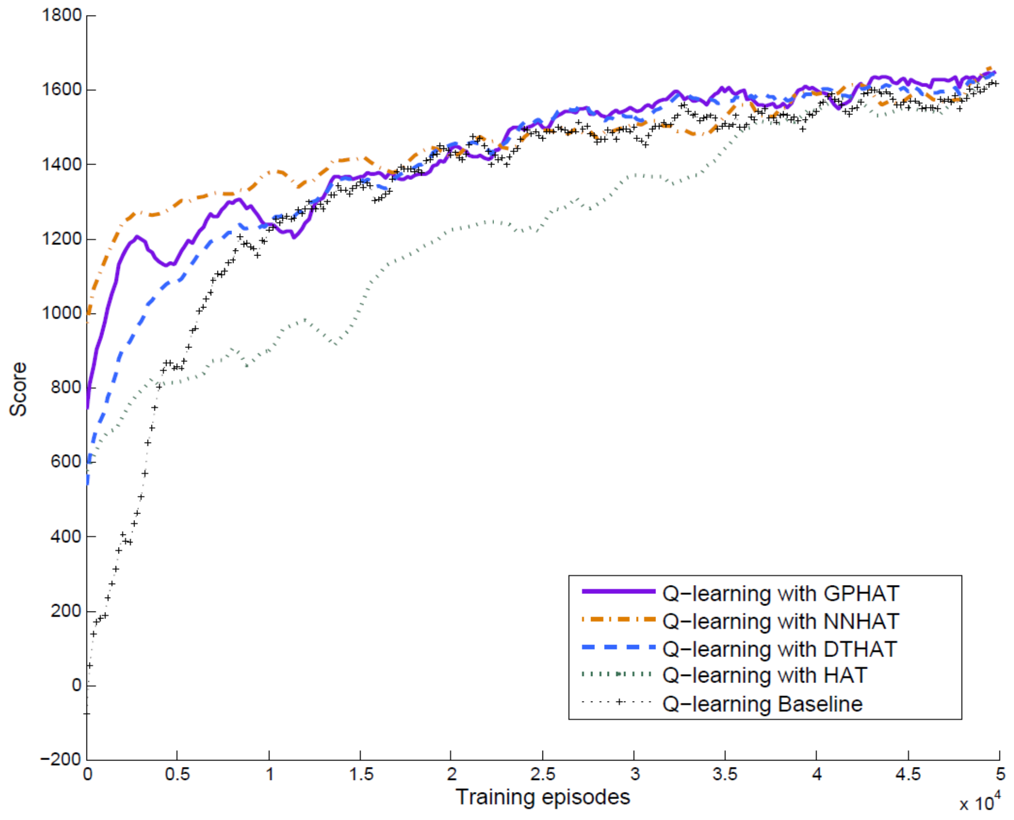
Probabilistic policy reuse:

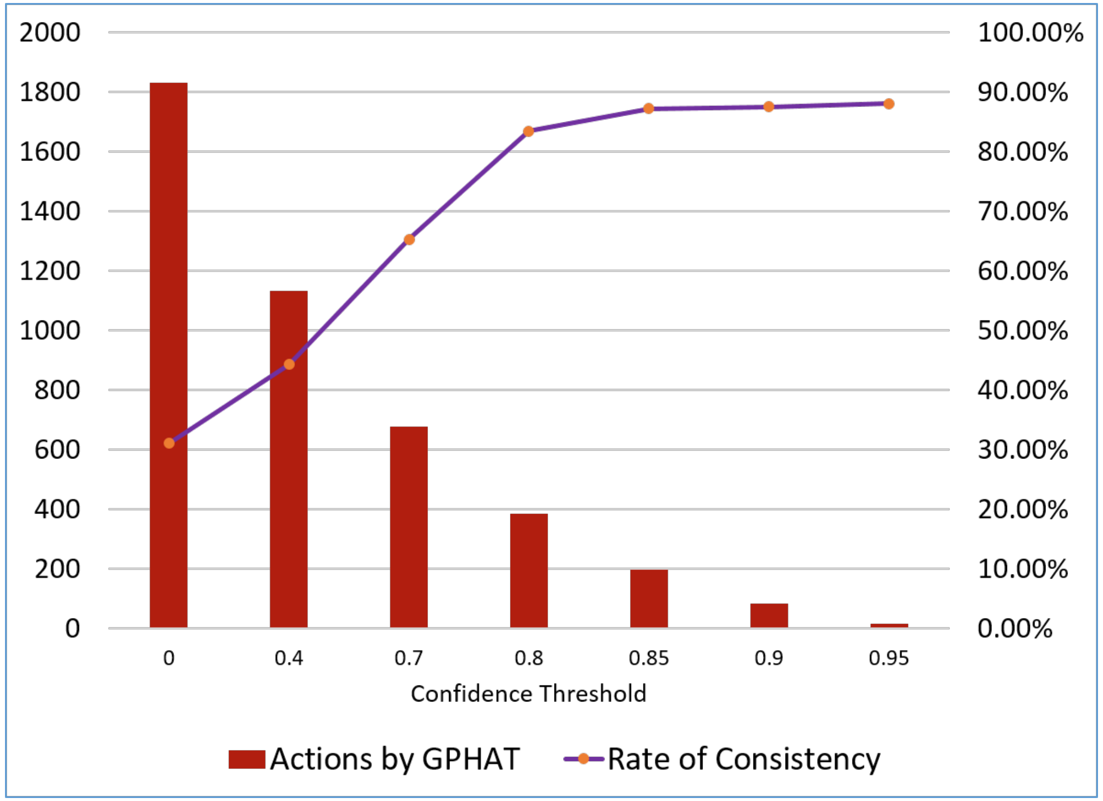
- Prior knowledge would be reused with a decaying probability



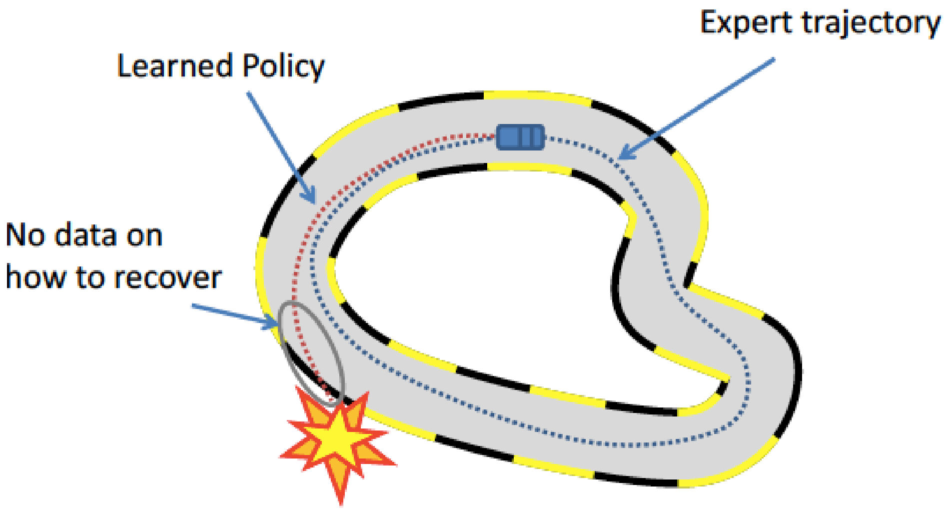
Performance Improvement

Mario domain





DAgger (Dataset Aggregation)



- Iterative algorithm
- Trains a stationary deterministic policy
- No regret algorithm in an online learning setting

[under reasonable assumptions, it] “must find a policy with good performance under the distribution of observations it induces in such sequential settings”

“A Reduction of Imitation Learning and Structured Prediction to No-Regret Online Learning”. Ross, Gordon, & Bagnell, 2011

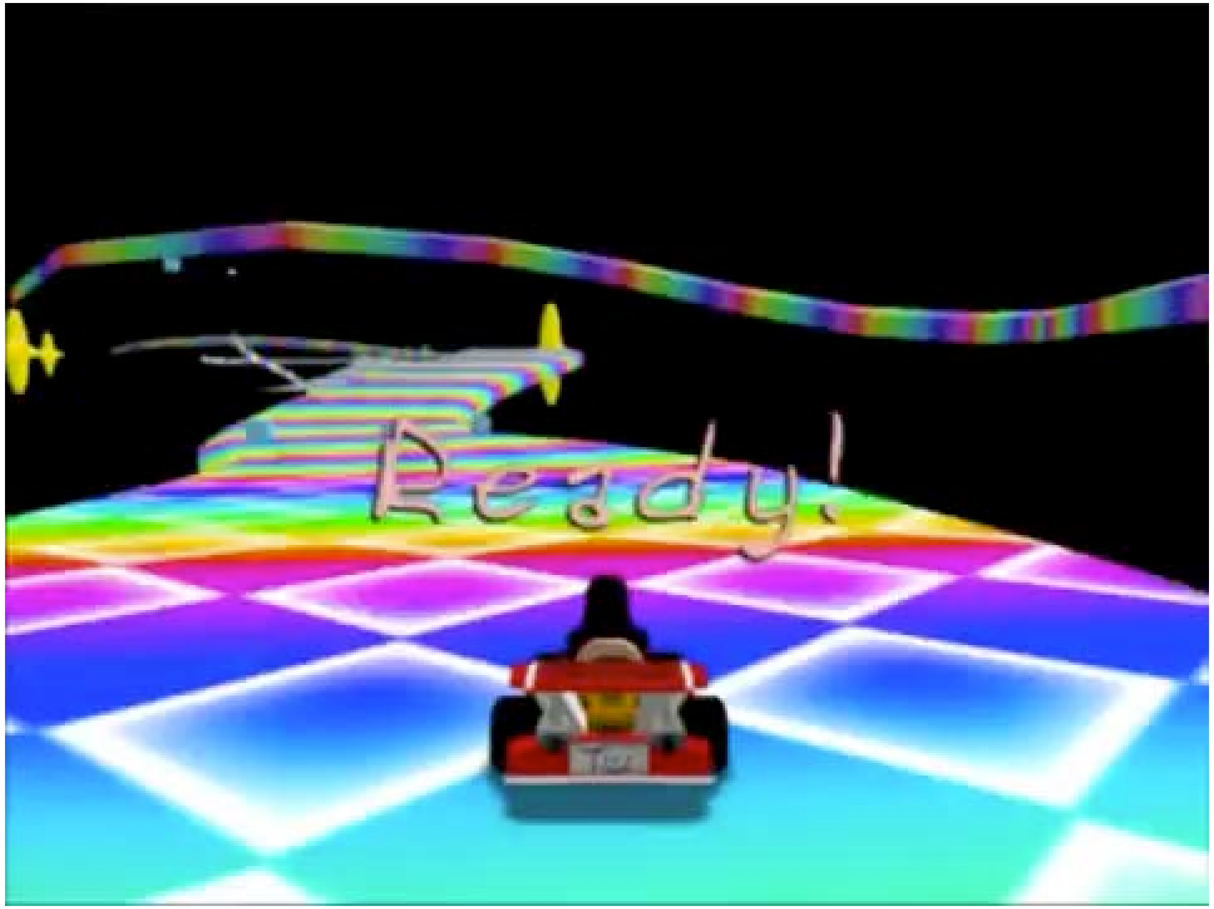
Initialize $\mathcal{D} \leftarrow \emptyset$.
Initialize $\hat{\pi}_1$ to any policy in Π.
for $i = 1$ **to** N **do**
  Let $\pi_i = \beta_i \pi^* + (1 - \beta_i)\hat{\pi}_i$.
  Sample T-step trajectories using π_i.
  Get dataset $\mathcal{D}_i = \{(s, \pi^*(s))\}$ of visited states by π_i and actions given by expert.
  Aggregate datasets: $\mathcal{D} \leftarrow \mathcal{D} \bigcup \mathcal{D}_i$.
  Train classifier $\hat{\pi}_{i+1}$ on $\mathcal{D}$.
end for
Return best $\hat{\pi}_i$ on validation.

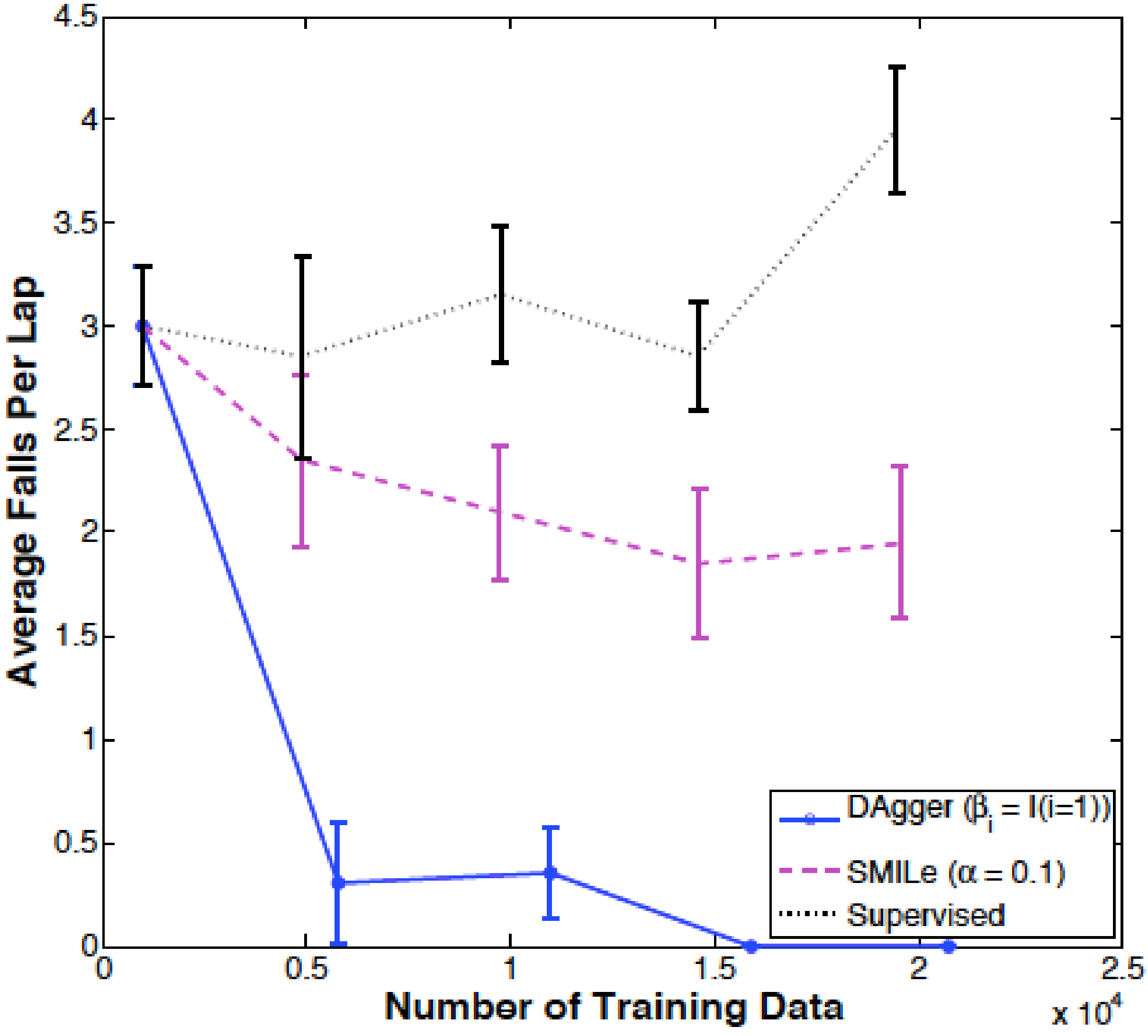
Algorithm 3.1: DAGGER Algorithm.

At the first iteration, it uses the expert's policy to gather a dataset of trajectories $\mathcal{D}$ and train a policy $\hat{\pi}_2$ that best mimics the expert on those trajectories. Then at iteration n, it uses $\hat{\pi}_n$ to collect more trajectories and adds those trajectories to the dataset $\mathcal{D}$. The next policy $\hat{\pi}_{n+1}$ is the policy that best mimics the expert on the whole dataset $\mathcal{D}$.

Insight:

1) Combine learned policy with novel human demos
2) Train over all of human demos
3) Learn about areas of the state space not initially reached





Super Tux Kart

AggreVaTe (Aggregate Values to Imitate)

- Expected future cost-to-go: $Q_t^{\pi}(s, a)$ of executing *a* in *s*, and then following π for *t-1* steps
- d_{π}^{t} distribution of states at time t induced by executing policy π
- Overall performance: $J(\pi) = \sum_{t=1}^{T} \mathbb{E}_{s \sim d_{\pi}^{t}}[C\big(s, \pi(s)\big)]$
- Observe expert perform task
- At uniformly random time, explores an action *a* in state *s*, and then get cost-to-go *Q* after performing this action
- Choose actions to minimize co-to-go instead of classification loss

"Reinforcement and Imitation Learning via Interactive No-Regret Learning". Ross & Bagnell, 2014

Algorithm 1 AGGREVATE: Imitation Learning with Cost-To-Go

Initialize $\mathcal{D} \leftarrow \emptyset$, $\hat{\pi}_1$ to any policy in Π.
for $i = 1$ **to** N **do**
  Let $\pi_i = \beta_i \pi^* + (1 - \beta_i)\hat{\pi}_i$ #Optionally mix in expert's own behavior.
  Collect m data points as follows:
  for $j = 1$ **to** m **do**
    Sample uniformly $t \in \{1, 2, \ldots, T\}$.
    Start new trajectory in some initial state drawn from initial state distribution
    Execute current policy π_i up to time $t - 1$.
    Execute some exploration action a_t in current state s_t at time t
    Execute expert from time $t + 1$ to T, and observe estimate of cost-to-go $\hat{Q}$ starting at time t
  end for
  Get dataset $\mathcal{D}_i = \{(s, t, a, \hat{Q})\}$ of states, times, actions, with expert's cost-to-go.
  Aggregate datasets: $\mathcal{D} \leftarrow \mathcal{D} \bigcup \mathcal{D}_i$.
  Train cost-sensitive classifier $\hat{\pi}_{i+1}$ on $\mathcal{D}$
    (Alternately: use any online learner on the data-sets $\mathcal{D}_i$ in sequence to get $\hat{\pi}_{i+1}$)
end for
Return best $\hat{\pi}_i$ on validation.

- Task performance of learned policies: related to regret on regression loss and the cost-to-go

- Task performance relates to the square root of the online learning regret and the regression regret of the best regressor in the class to the Bayes-optimal regressor on this training data

- Potential drawback: “any method relying on cost-to-go estimates can be impractical as collecting each estimate for a single state-action pair may involve executing an entire trajectory”

LfD + Shaping Rewards: Similarity Based Shaping

- RL + LfD: RLFD
- Want high potential function when action was demonstrated nearby
- Given demonstrations & similarity/distance function:
  - Create potential shaping function on the fly
- Think: placing Gaussians on demonstrated (s,a)
  - Local reward shaping

“Reinforcement Learning from Demonstration through Shaping”. Brys, Harutyunyan, Suay, Chernova, Taylor, and Nowé, 2015.

LfD + Shaping Rewards: Similarity Based Shaping

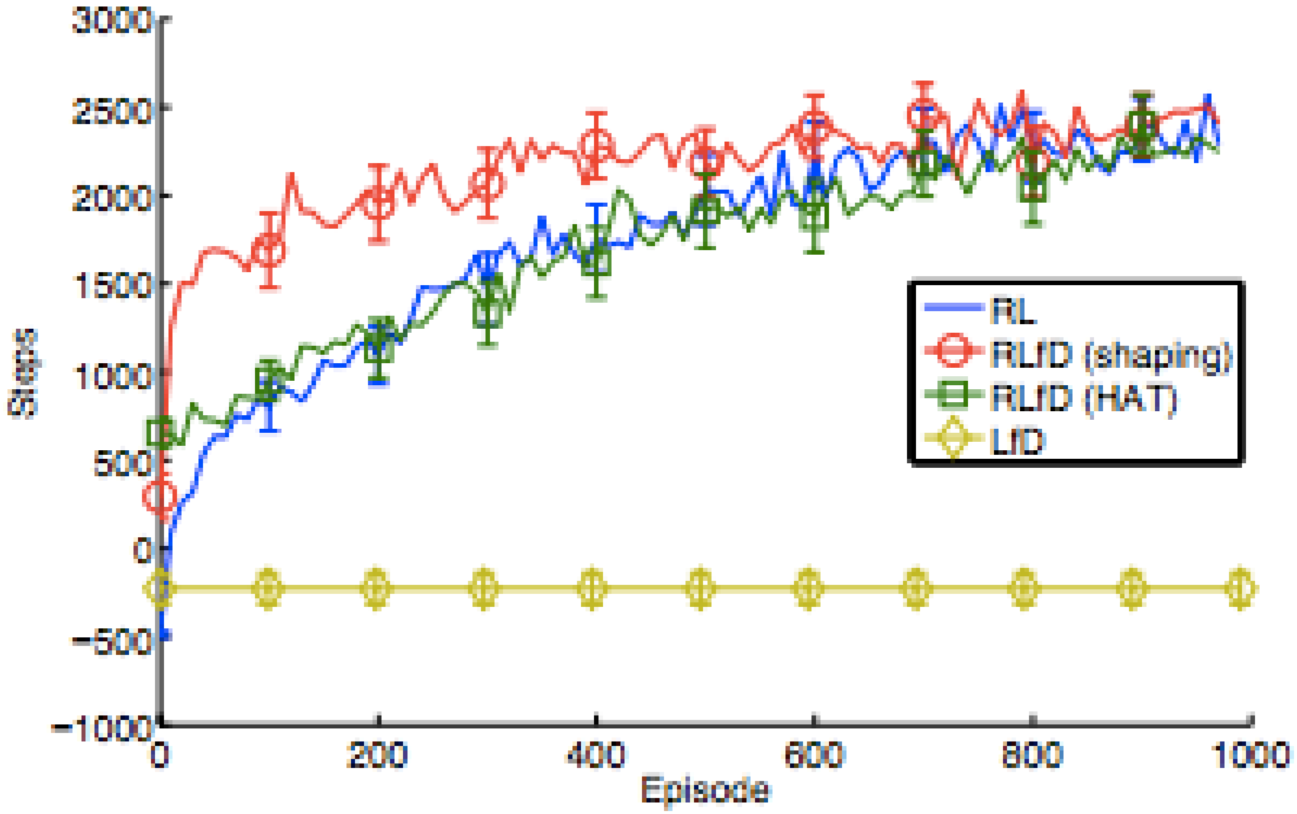
- Define similarity measure between states and actions

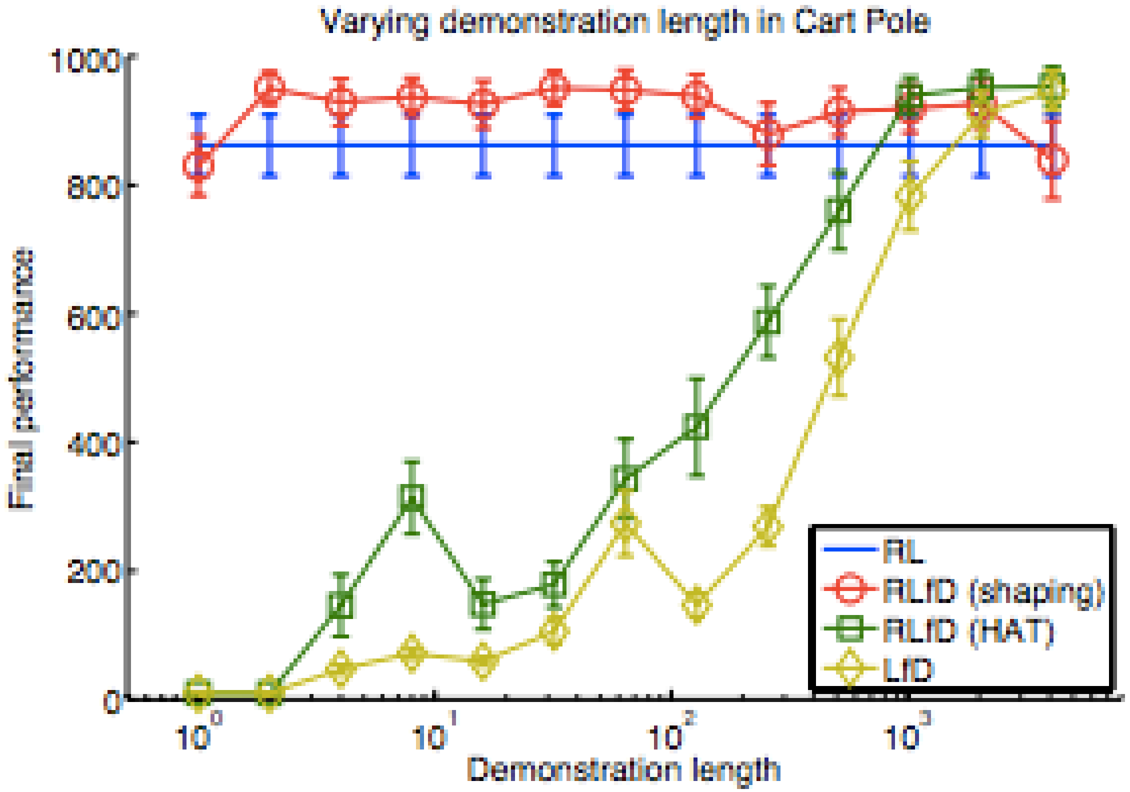
$$sim(s,a,s^d,a^d,\Sigma) = \begin{cases} 0 & \text{if } a \neq a^d \\ e^{\left(-\frac{1}{2}(s-s^d)^T\Sigma^{-1}(s-s^d)\right)} & \text{if } a = a^d \end{cases}$$

- Set potential to highest similarity among demonstrated samples

$$\Phi(s,a) = \max_{(s^d,a^d)} sim(s,a,s^d,a^d,\Sigma)$$

- RL ($Q(\lambda)$-learning)
- RLfD ($Q(\lambda)$-learning+shaping)
- RLfD ($Q(\lambda)$-learning+HAT)
- LfD (C4.5 decision tree classifier [Quinlan, 1993])





Learning on Mario from 1 demonstration

Inverse Reinforcement Learning

MDP/R

- “Algorithms for Inverse Reinforcement Learning”. Ng & Russell, 2000.
- “Apprenticeship Learning via Inverse Reinforcement Learning”. Abbeel & Ng, 2004.

Model-free IRL:

- “Relative Entropy Inverse Reinforcement Learning”. Boularias, Kober, & Peters, 2011.

IRL + Shaping: Static IRL Shaping (SIS)

- Collect demonstrations: $(s_1,a_1,s_2,s_2,\ldots)$
- Learn reward function over states using IRL
- Use new reward function as potential-based shaping reward over states:
  - $F(s,a,s') = \gamma\Phi(s')-\Phi(s)$
  - $R' = R+F$
- Potential function does not change over time
- The effect of shaping is that the agent's exploration is less random and the agent is biased towards states with high potential

"Learning from Demonstration for Shaping through Inverse Reinforcement Learning". Suay, Brys, Taylor, & Chernova, 2016

IRL + Shaping: Dynamic IRL Shaping (SIS)

- Collect demonstrations: $(s_1,a_1,s_2,a_2,\ldots)$
- Learn reward function using states and actions IRL
- Use dynamic shaping: $F(s,a,t,s',a',t') = \gamma\Phi(s',a',t')-\Phi(s,a,t)$

- Learn secondary Q-function online for potential function
  - $\Phi_2(s,a) \leftarrow \Phi_2(s,a) + \alpha_2(r_{IRL}(s) + \gamma\Phi_2(s',a')-\Phi_2(s,a))$
  - Q-function gets updated online after each observation
- Now use this (changing) potential-based function:
  - $F = \gamma\Phi_2(s',a')-\Phi_2(s,a)$
  - $R' = R+F$

- Autonomous Learning: RL
- Demonstration + RL
  - action selection (time to go)
  - shaping reward
  - IRL: shaping reward
- Learning from human feedback
  - Treat as environment reward
  - Treat as return
  - Return + RL
  - Treat as categorical feedback regrading policy

Learning *Directly* from Human Reward

- Sophie's Kitchen
- Human trainer can award a scalar reward signal r = [−1, 1]

Algorithm 1 Q-Learning with Interactive Rewards:
s = last state, s' = current state, a = last action, r = reward

1: **while** learning **do**
2: a = random select weighted by $Q[s, a]$ values
3: execute a, and transition to s'
(small delay to allow for human reward)
4: sense reward, r
5: update values:

$$Q[s,a] \leftarrow Q[s,a] + \alpha(r + \gamma(max_{a'} Q[s',a']) - Q[s,a])$$

6: **end while**

"Reinforcement Learning with Human Teachers: Evidence of Feedback and Guidance with Implications for Learning Performance". Thomaz & Breazeal, 2006.

Learning *Directly* from Human Reward

- Anticipatory Guidance Rewards

- "Even though our instructions clearly stated that communication of both general and object specific rewards, we found many people assumed that object specific rewards were future directed messages or guidance for the agent. Several people mentioned this in the interview, and we also find behavioral evidence in the game logs."

- They provide
  1) anticipatory reward (direct future) &
  2) feedback for past actions

Algorithm 2 Interactive Q-Learning modified to incorporate interactive human guidance in addition to feedback.

while learning **do**
    while waiting for guidance **do**
        if receive human guidance message **then**
            $g = \textit{guide-object}$
        end if
    end while
    if received guidance **then**
        $a =$ random selection of actions containing g
    else
        $a =$ random selection weighted by $Q[s, a]$ values
    end if
    execute a, and transition to s'
    (small delay to allow for human reward)
    sense reward, r
    update values:

$$Q[s, a] \leftarrow Q[s, a] + \alpha(r + \gamma(max_{a'} Q[s', a']) - Q[s, a])$$

end while

Learning from Human Rewards: Interactive Shaping

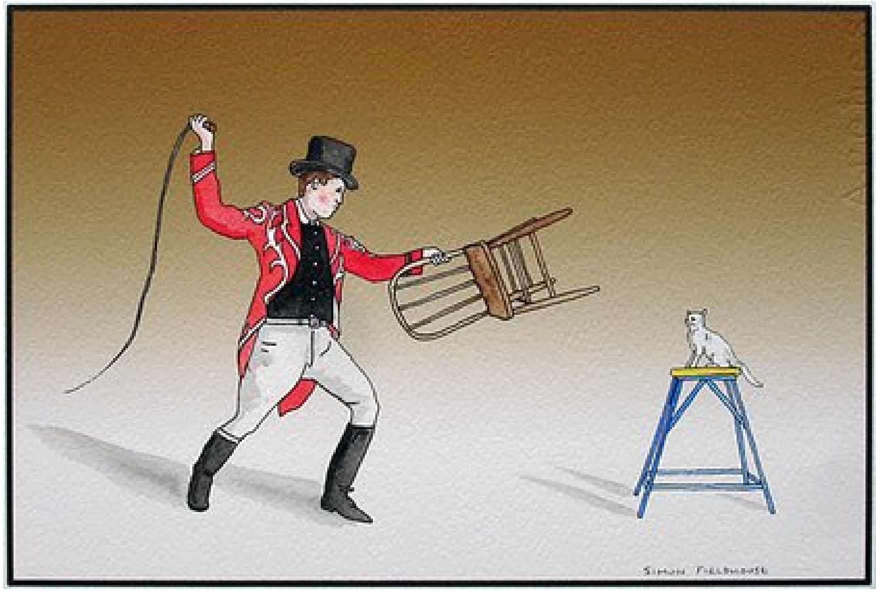
TAMER

Key insight: trainer evaluates behavior using model of its long-term quality

Learn a model of human reinforcement

$$H : S \times A \to \mathbb{R}$$

Directly exploit the model to determine action
Also, can combine with MDP's reward



http://www.cincinnatireview.com/blog/tag/lion-tamer/

http://www.bradknox.net/projects/

TAMER Learning Tetris

Initial Training

After 2 games of Training

TAMER+RL

- 2 settings
  - Sequential
  - Simultaneous

- **Reward shaping**: $R'(s,a) = R(s,a) + (\beta * \hat{H}(s,a))$
- **Q augmentation**: $Q'(s,a) = Q(s,a) + (\beta * \hat{H}(s,a))$
- **Action biasing**: $Q'(s,a) = Q(s,a) + (\beta * \hat{H}(s,a))$ *only during action selection*
- **Control sharing**: $P(a = argmax_a[\hat{H}(s,a)]) = min(\beta, 1)$. *Otherwise use base RL agent's action selection mechanism.*

- Important points:
  - Decaying influence
  - Eligibility traces for reward

"Combining manual feedback with subsequent MDP reward signals for reinforcement learning". Knox & Stone, 2010.
"Reinforcement Learning from Simultaneous Human and MDP Reward". Knox & Stone, 2012.

Motivation: Dog Training

- Teach dog to sit & shake

Policy $\begin{cases} \text{“Sit”} \rightarrow \\ \text{“Shake”} \rightarrow \end{cases}$

- Mapping from observations to actions
- Feedback: {Bad Dog, Good Boy}

History of Evidence

- Feedback history *h*
- Observation: “sit”, Action:  , Feedback: “Bad Dog”
- Observation: “sit”, Action:  , Feedback: “Good Boy”
- …
- Really make sense to assign **numeric** rewards to these?

Bayesian Framework

- Trainer desires policy $\lambda*$
- h_t is the training history at time t
- Find MAP hypothesis of $\lambda*$:

$$\underset{\lambda}{\operatorname{argmax}}\, p(\lambda^* = \lambda | h_t) = \underset{\lambda}{\operatorname{argmax}}\, p(h_t | \lambda^* = \lambda) p(\lambda^* = \lambda)$$

Prior distribution over policies

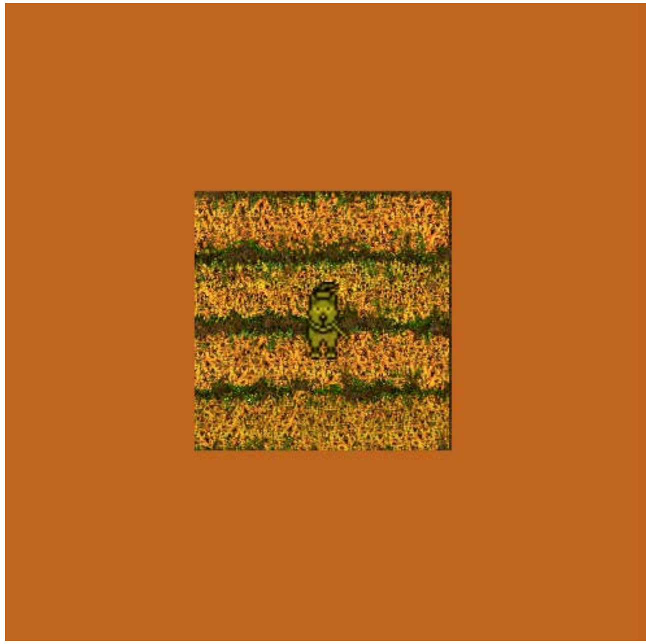
Model of training process

“Learning behaviors via human-delivered discrete feedback: modeling implicit feedback strategies to speed up learning”. Loftin, Peng, MacGlashan, Littman, Taylor, Huang, & Roberts, 2015

Strategy-Aware Bayesian Learning (SABL)

Assuming trainer feedback is given according to a probabilistic model (with known μ^+, μ^- and ϵ)

- action was correct, with error probability ϵ
- withhold or give explicit feedback, with probability μ^+ and μ^-



Compute a maximum likelihood estimate of the target policy λ, given a training history h:

$$\lambda^* = argmax_\lambda Pr[h|\lambda, \mu^+, \mu^-, \epsilon]$$

Strategy-Aware Bayesian Learning (SABL)

To a strategy-aware learner, the lack of feedback can be as informative as explicit feedback

No feedback?

That is not what I want, try something else!

Keep going and you will get reward eventually!

Infer Neutral

- Try to learn what no-reward (μ^+ & μ^-) means
- Don't assume they're balanced

- Many trainers don't use punishment
  - Neutral feedback = punishment

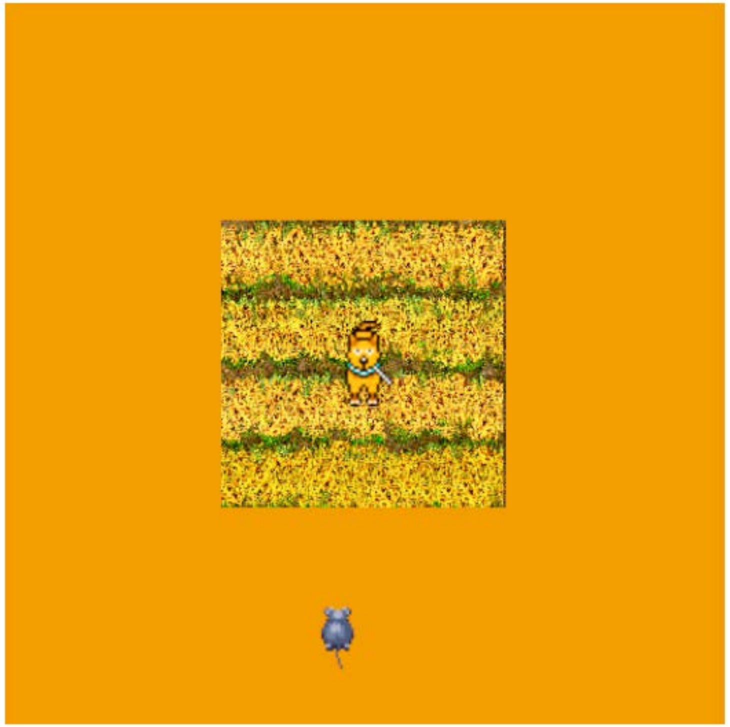
- Some don't use reward
  - Neutral feedback = reward

How Humans Reward

- Turkers & Dog Training Enthusiasts
- Explicitly reward good behavior? R+
- Explicitly punish bad behavior? P+
- Stay consistent over time?

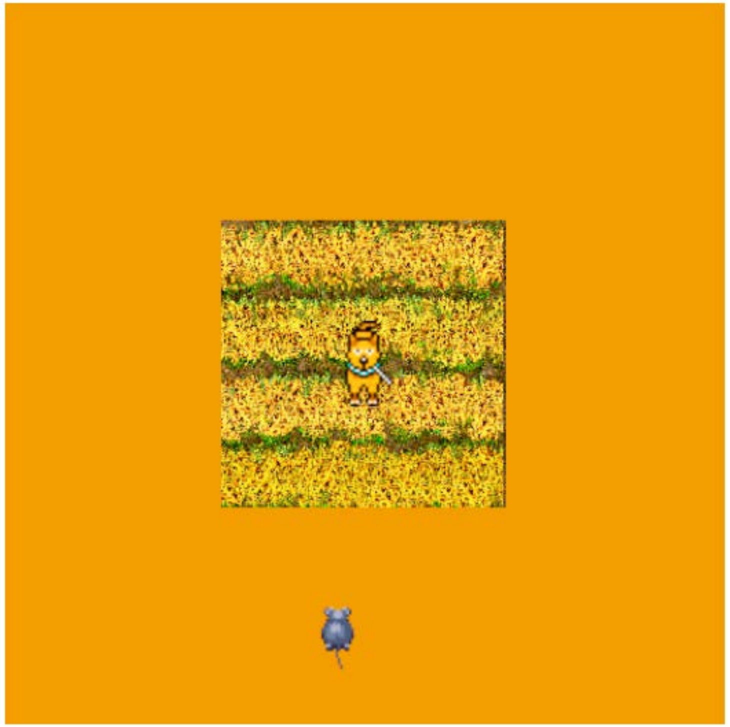
	P+	P−
R+	93	125
R−	6	3

begin \ end	R+/P+	R+/P−	R−/P+	R−/P−
R+/P+	65	4	2	0
R+/P−	10	52	1	1
R−/P+	2	1	4	1
R−/P−	0	0	0	1



Protect the Field

How Humans Reward



Protect the Field

- Get the battery

- Eat the bird

- Point towards the box

Policy Shaping

- Simulated Oracle: theoretical analysis
- Combines human feedback with RL
- Positive and negative trainer feedback = discrete communication that depends on trainer's target policy
- Feedback can be correct [consistent] with some probability *C* and human will provide feedback with some likelihood *L*

"Policy shaping: Integrating human feedback with reinforcement learning". Griffith, Subramanian, Scholz, Isbell, & Thomaz, 2013

Policy Shaping

Difference between number of "right" and "wrong" labels: $\Delta_{s,a}$

Prob s,a is optimal (binomial distribution): $$\frac{\mathcal{C}^{\Delta_{s,a}}}{\mathcal{C}^{\Delta_{s,a}} + (1-\mathcal{C})^{\Delta_{s,a}}}$$

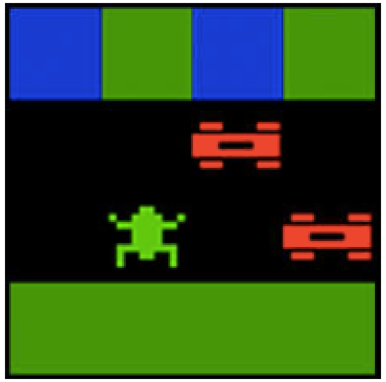
Combine probabilities of different actions based on learned Q-values (Bayesian Q-Learning) and critique advice $$\frac{P_q(a)P_c(a)}{\sum_{a\in A} P_q(a)P_c(a)}$$

Very similar to Q-learning when

1. Small amount of human critique
2. Critique equal among many s/a pairs
3. Human is right roughly half the time
   (*C* is close to 0.5)

Pac-Man

Frogger



Policy Shaping

- 2nd paper: focus on human participants
- Participants: shown videos of recorded trajectories
- Goals:
  - Humans vs. Oracle
  - Value of silence
- Provide positive or negative feedback
- Error rate and assumptions re: +/- set by fixed params

"Policy Shaping With Human Teachers". Cederborg, Grover, Isbell & Thomaz, 2015.

Policy Shaping

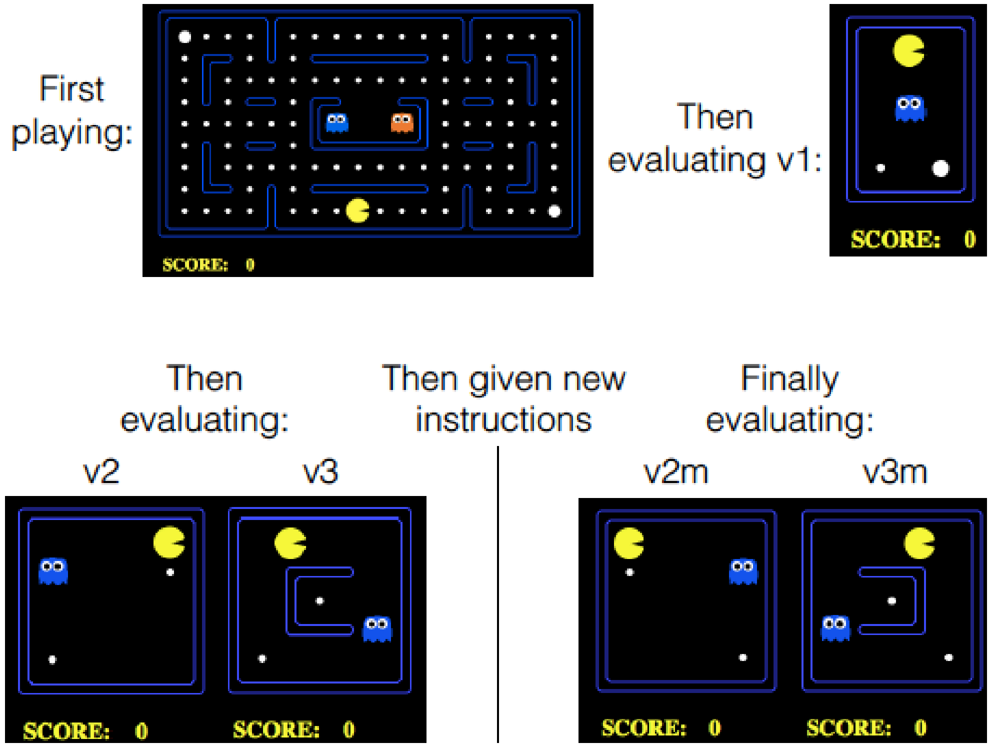


Figure 1: Each teacher first plays the large board to the top left. Then evaluates videos v1, v2 and v3. New instructions are given based on what group the teacher has been assigned to, then v2m and v3m are evaluated.

Investigate:

- Humans can provide good data for shaping
- People have inherent bias regarding silence
- Can manipulate meaning of silence

Experiments

- Oracle: simulated teacher
- Human-unbiased: a human teacher provides action critiques, with no instruction about the meaning of silence.
- Human-positive bias: instruction that silence is positive
- Human-negative bias: with instruction that silence is negative

Policy Shaping

Primary result: Humans could give useful feedback

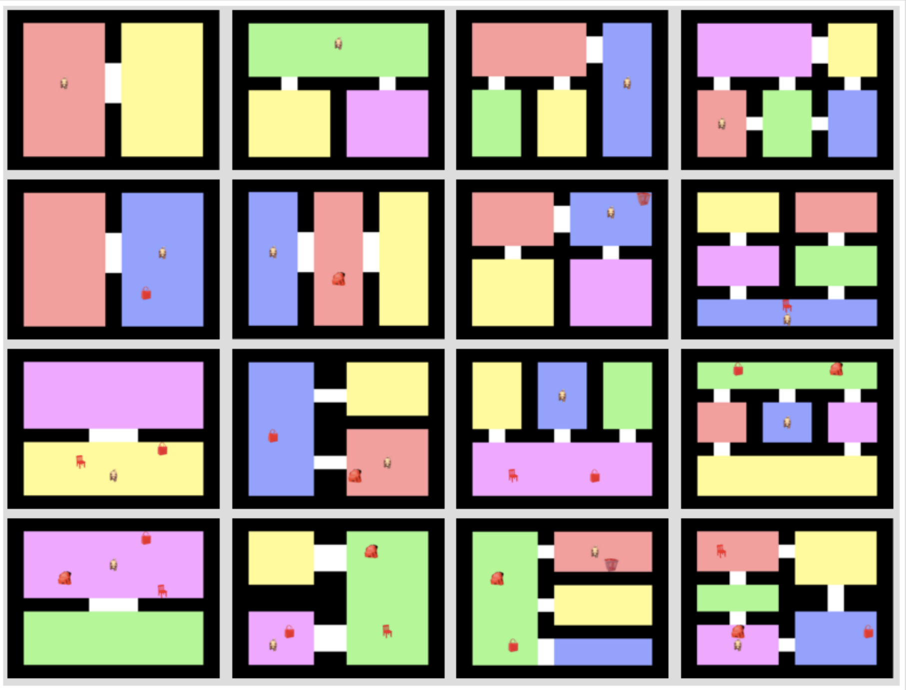
- “Even when giving instructions biasing silence towards bad, it is still better to assume that silence means good.”
- “It could be that people tend to mean silence as good”
- “However, to fully convince ourselves of this we would need to experiment on a variety of domains with different positive/ negative biases”

Aside: Learning lower-level skills

- (e.g., Dynamic Motion Primitives)
- Particularly important in Robotics
- “Reinforcement Learning in Robotics: A Survey”. Kober, Bagnell, & Peters, 2013.

Open Questions: 1/2

- Two-way communication
  - Asking for help
  - Human knows what robot knows
    - Robot knows human knows what robot knows
      - Human knows robot knows human knows what robot knows…
- Steer human towards useful feedback
  - Reciprocal interaction
  - Human effective at shaping a given agent.
  - "Eliciting good teaching from humans for machine learners". Cakmak & Thomaz, 2014
  - "A Need for Speed: Adapting Agent Action Speed to Improve Task Learning from Non-Expert Humans". Peng+, 2016.
- Curriculum Learning



Open Questions: 2/2

- Best way to teach people to teach?
- Different modalities
  - LfD vs. LfF
  - “Understanding Human Teaching Modalities in Reinforcement Learning Environments: A Preliminary Report”. Knox, Taylor, & Stone. 2011
- Treating experts of different quality differently
- Testing with *normal people*
  - “A practical comparison of three robot learning from demonstration algorithm”. Suay, Toris, & Chernova, 2012.
- Crowdsourcing ?
  - “The Robot Management System: A Framework for Conducting Human-Robot Interaction Studies Through Crowdsourcing”. Toris, Kent, & Chernova, 2014.

References

- Sutton & Barto, “Reinforcement Learning: An Introduction” https://webdocs.cs.ualberta.ca/~sutton/book/ebook/the-book.html
- Littman & Isbell Udacity course, “Reinforcement Learning” https://classroom.udacity.com/courses/ud600/
- Szepesvári, “Algorithms for Reinforcement Learning” https://sites.ualberta.ca/~szepesva/RLBook.html
- “Integrating Reinforcement Learning with Human Demonstrations of Varying Ability”. Taylor, Suay, & Chernova, 2011
- “Improving Reinforcement Learning with Confidence-Based Demonstrations”. Wang & Taylor, 2017
- “A Reduction of Imitation Learning and Structured Prediction to No-Regret Online Learning”. Ross, Gordon, & Bagnell, 2011
- “Reinforcement and Imitation Learning via Interactive No-Regret Learning”. Ross & Bagnell, 2014
- “Reinforcement Learning from Demonstration through Shaping”. Brys, Harutyunyan, Suay, Chernova, Taylor, and Nowé, 2015.
- “Algorithms for Inverse Reinforcement Learning”. Ng & Russell, 2000.
- “Apprenticeship Learning via Inverse Reinforcement Learning”. Abbeel & Ng, 2004.
- Relative Entropy Inverse Reinforcement Learning”. Boularias, Kober, & Peters, 2011.
- “Learning from Demonstration for Shaping through Inverse Reinforcement Learning”. Suay, Brys, Taylor, & Chernova, 2016
- “Reinforcement Learning with Human Teachers: Evidence of Feedback and Guidance with Implications for Learning Performance”. Thomaz & Breazeal, 2006.
- “Combining manual feedback with subsequent MDP reward signals for reinforcement learning”. Knox & Stone, 2010.
- “Reinforcement Learning from Simultaneous Human and MDP Reward”. Knox & Stone, 2012.
- “Learning behaviors via human-delivered discrete feedback: modeling implicit feedback strategies to speed up learning”. Loftin, Peng, MacGlashan, Littman, Taylor, Huang, & Roberts, 2015
- “Policy shaping: Integrating human feedback with reinforcement learning”. Griffith, Subramanian, Scholz, Isbell, & Thomaz, 2013
- “Policy Shaping With Human Teachers”. Cederborg, Grover, Isbell & Thomaz, 2015.
- “Reinforcement Learning in Robotics: A Survey”. Kober, Bagnell, & Peters, 2013.