New Applications of Deep Generative Models

Jiaming Song
Stanford University

11/19/2018

Deep Generative Models

Deep Generative Models

- Images (BigGAN, Glow)

Deep Generative Models

- Images (BigGAN, Glow)

Deep Generative Models

- Images (BigGAN, Glow)

- Audio (WaveNet)

Topics

- Imitation Learning
  - Distribution matching view of imitation learning
  - Multi-Agent Generative Adversarial Imitation Learning
- Fair Representation Learning
  - Information-theoretic notions on latent variable generative models
  - Learning Controllable Fair Representations

Multi-Agent Generative Adversarial Imitation Learning

Jiaming Song

w/ Hongyu Ren, Dorsa Sadigh and Stefano Ermon

Stanford University

Reinforcement Learning

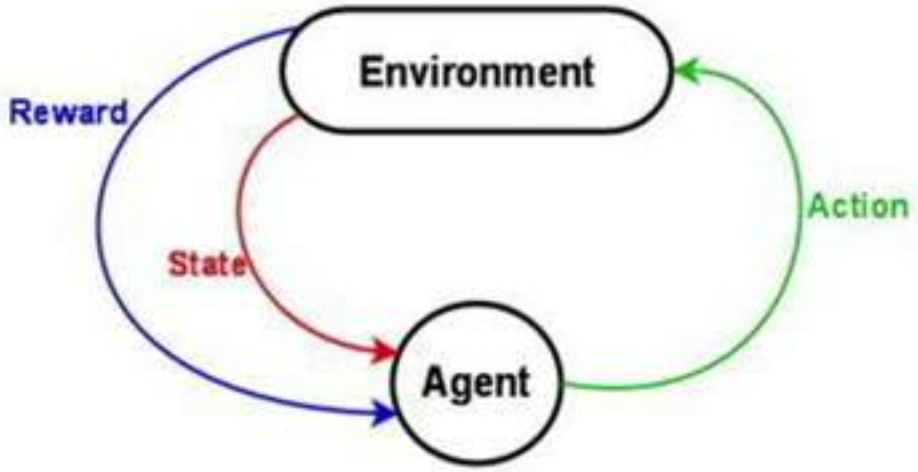
- Goal: Learn policies

Reinforcement Learning

- Goal: Learn policies
- High-dimensional, raw observations

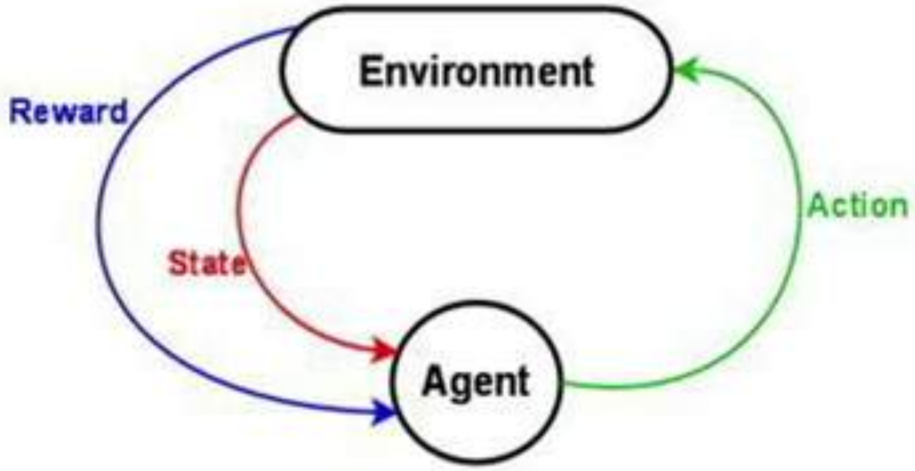
Reinforcement Learning

- Goal: Learn policies
- High-dimensional, raw observations



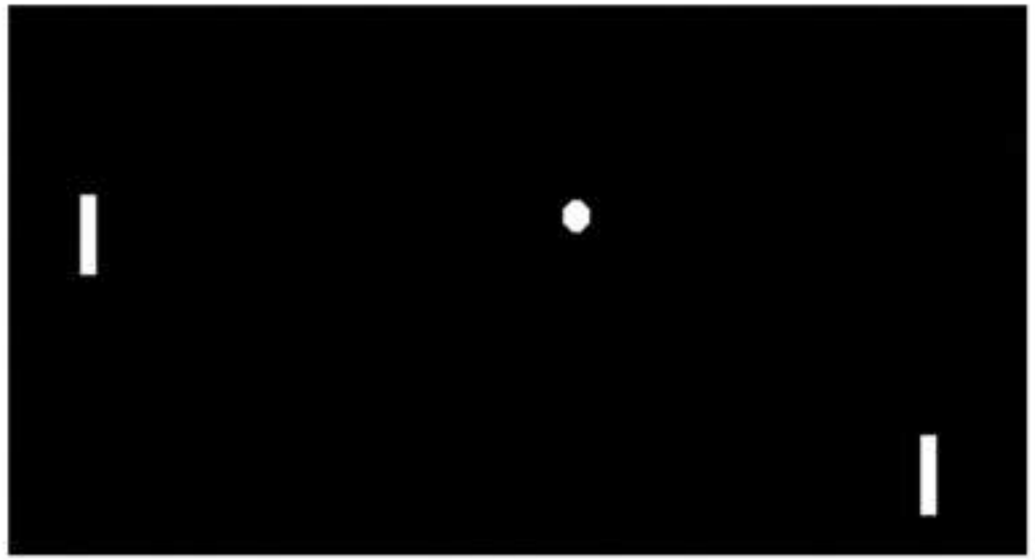
Reinforcement Learning

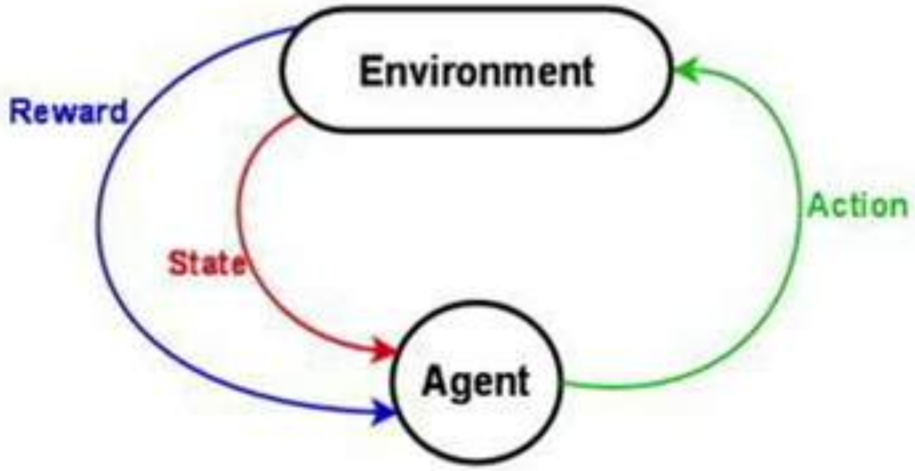
- Goal: Learn policies
- High-dimensional, raw observations

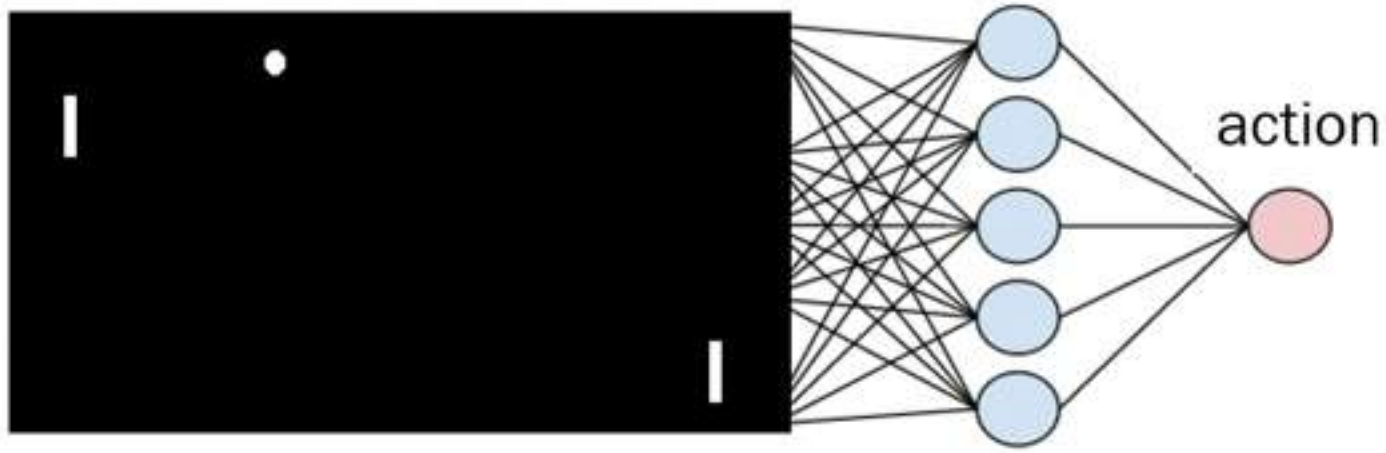


Reinforcement Learning

- Goal: Learn policies
- High-dimensional, raw observations

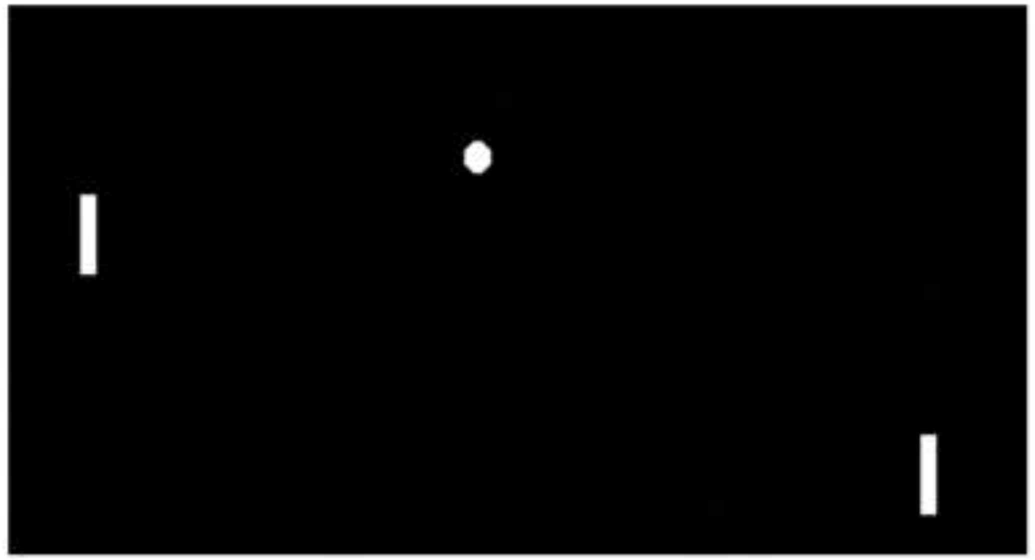


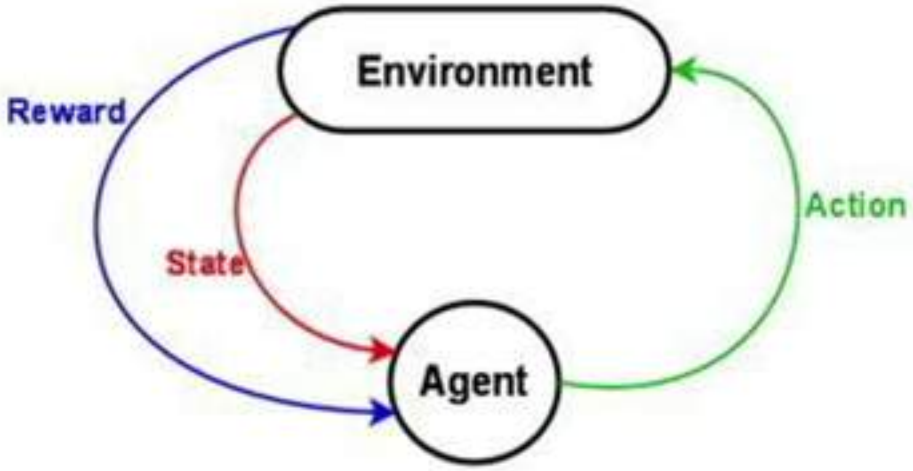




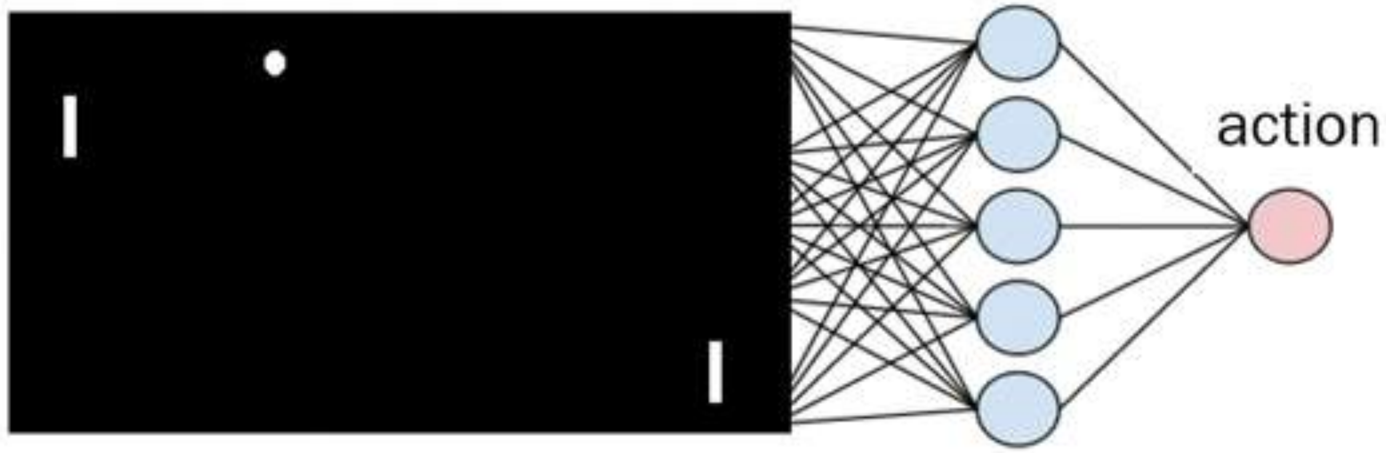
Reinforcement Learning

- Goal: Learn policies
- High-dimensional, raw observations





RL needs cost signal



Imitation

(Ng and Russell, 2000), (Abbeel and Ng, 2004; Syed and Schapire, 2007), (Ratliff et al., 2006), (Ziebart et al., 2008), (Kolter et al., 2008), (Finn et al., 2016), etc.

Imitation

Input: expert behavior generated by π_E

$$\{(s_0^i, a_0^i, s_1^i, a_1^i, \ldots)\}_{i=1}^n \sim \pi_E$$

(Ng and Russell, 2000), (Abbeel and Ng, 2004; Syed and Schapire, 2007), (Ratliff et al., 2006), (Ziebart et al., 2008), (Kolter et al., 2008), (Finn et al., 2016), etc.

Imitation

Input: expert behavior generated by π_E

$$\{(s_0^i, a_0^i, s_1^i, a_1^i, \dots)\}_{i=1}^n \sim \pi_E$$

Goal: learn *cost function (reward) or policy*

(Ng and Russell, 2000), (Abbeel and Ng, 2004; Syed and Schapire, 2007), (Ratliff et al., 2006), (Ziebart et al., 2008), (Kolter et al., 2008), (Finn et al., 2016), etc.

Problem setup

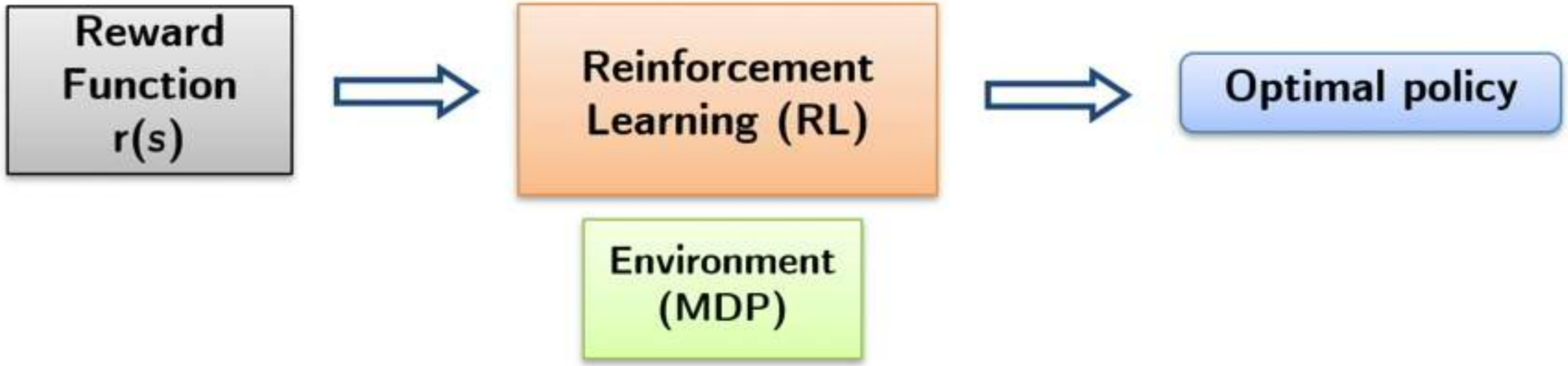
$$\mathrm{RL}(r) = \arg\max_{\pi \in \Pi} H(\pi) + \mathbb{E}_\pi[r(s,a)]$$

Problem setup

$$\mathrm{RL}(r) = \arg\max_{\pi \in \Pi} H(\pi) + \mathbb{E}_\pi[r(s,a)]$$

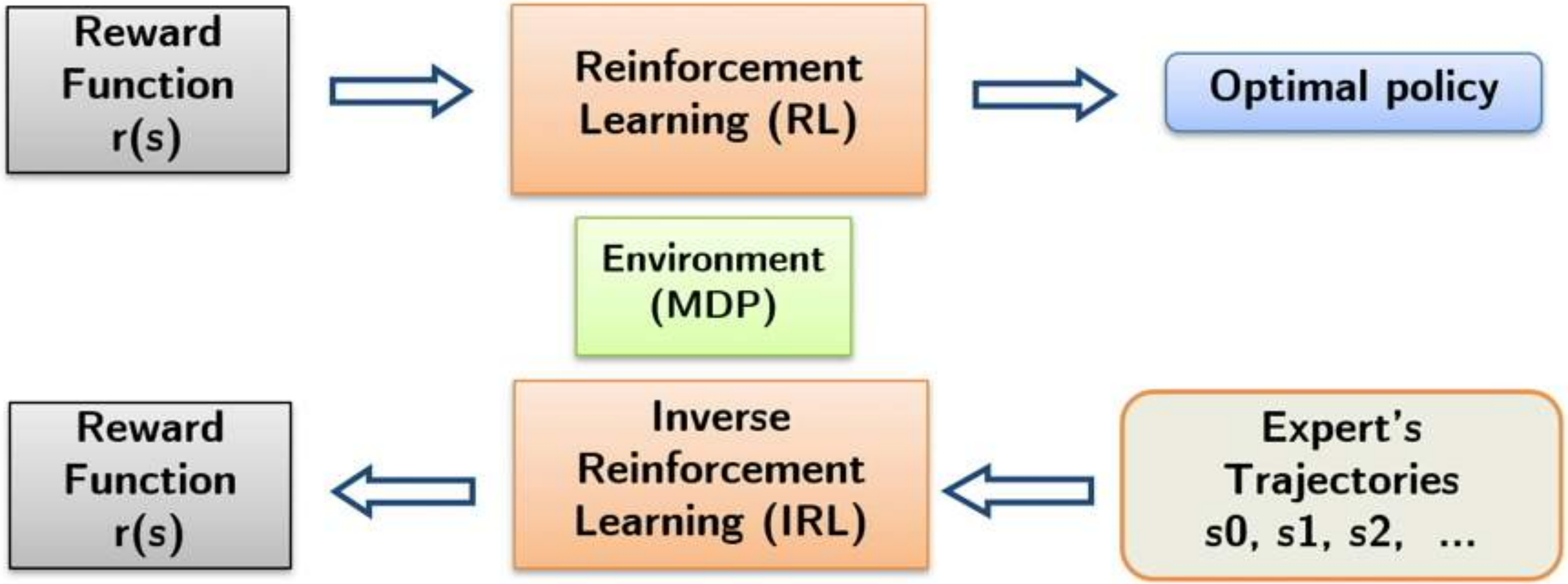
Problem setup

$$\mathrm{RL}(r) = \arg\max_{\pi \in \Pi} H(\pi) + \mathbb{E}_\pi[r(s,a)]$$



Problem setup

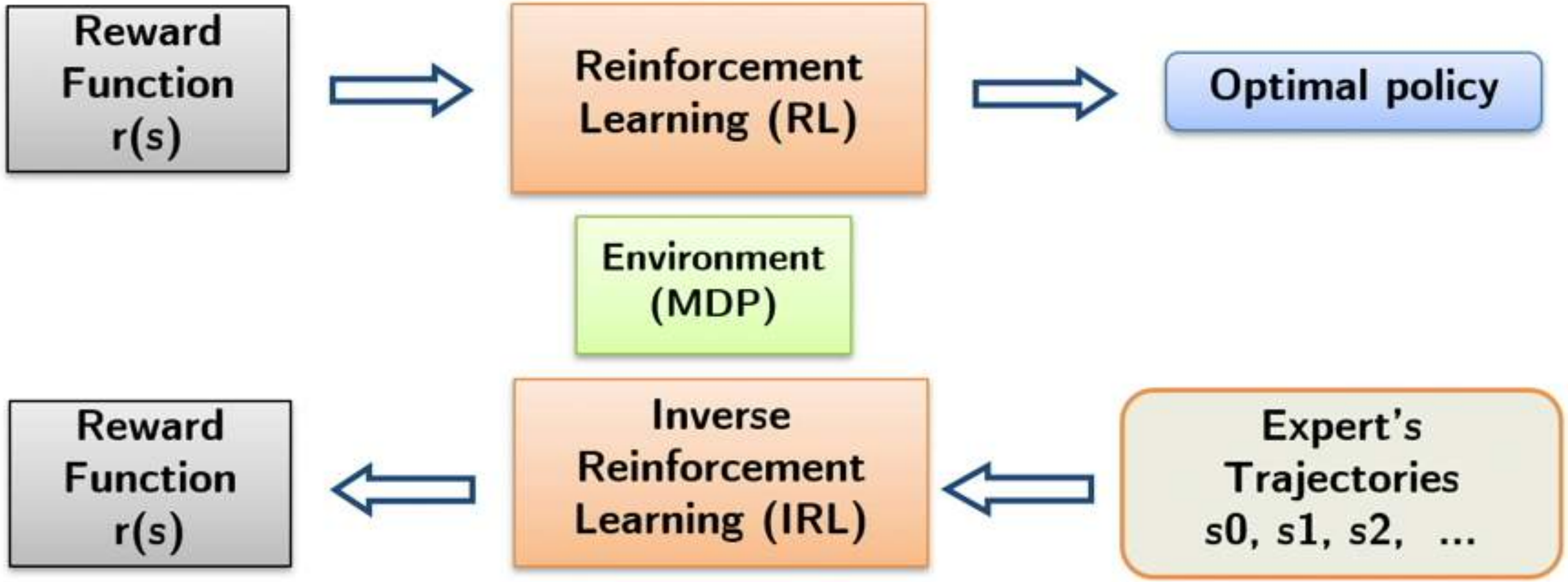
$$\mathrm{RL}(r) = \arg\max_{\pi \in \Pi} H(\pi) + \mathbb{E}_\pi[r(s,a)]$$



(Ziebart et al., 2010; Rust 1987)

Problem setup

$$\mathrm{RL}(r) = \arg\max_{\pi \in \Pi} H(\pi) + \mathbb{E}_{\pi}[r(s,a)]$$

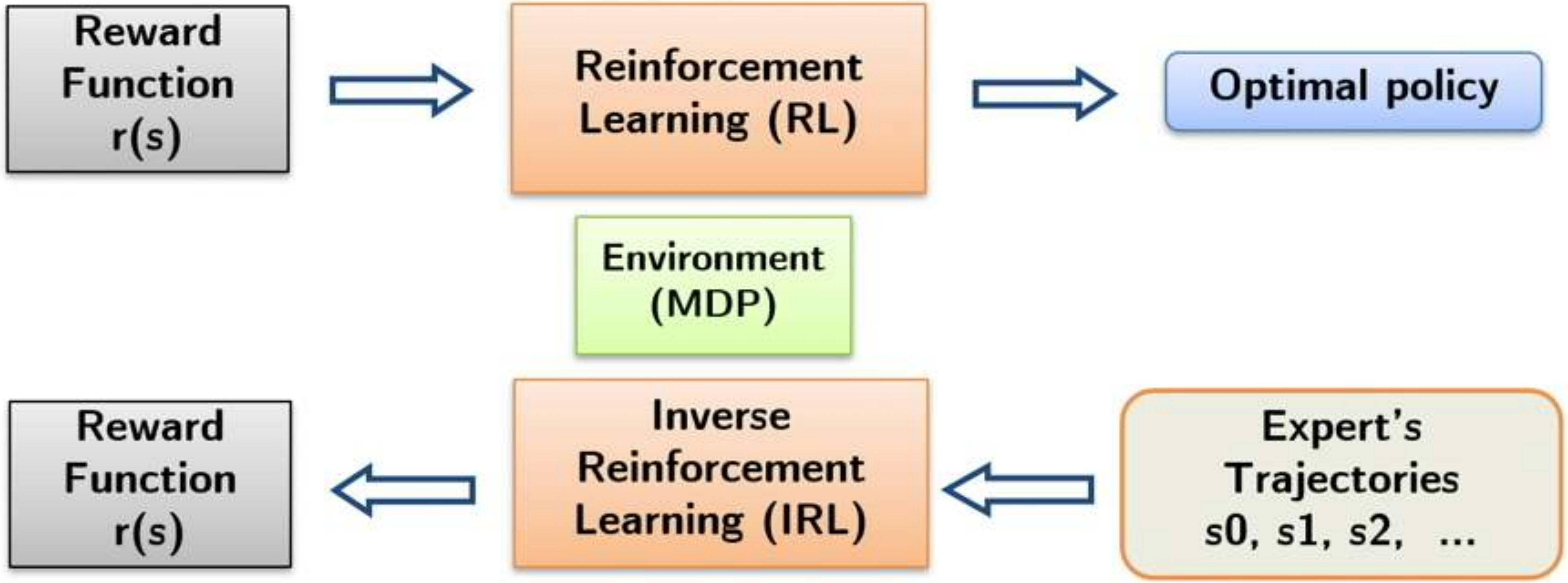


$$\mathrm{IRL}(\pi_E) = \arg\max_{r \in \mathbb{R}^{\mathcal{S} \times \mathcal{A}}} \mathbb{E}_{\pi_E}[r(s,a)] - \left(\max_{\pi \in \Pi} H(\pi) + \mathbb{E}_{\pi}[r(s,a)] \right)$$

(Ziebart et al., 2010; Rust 1987)

Problem setup

$$\mathrm{RL}(r) = \arg\max_{\pi \in \Pi} H(\pi) + \mathbb{E}_\pi[r(s,a)]$$



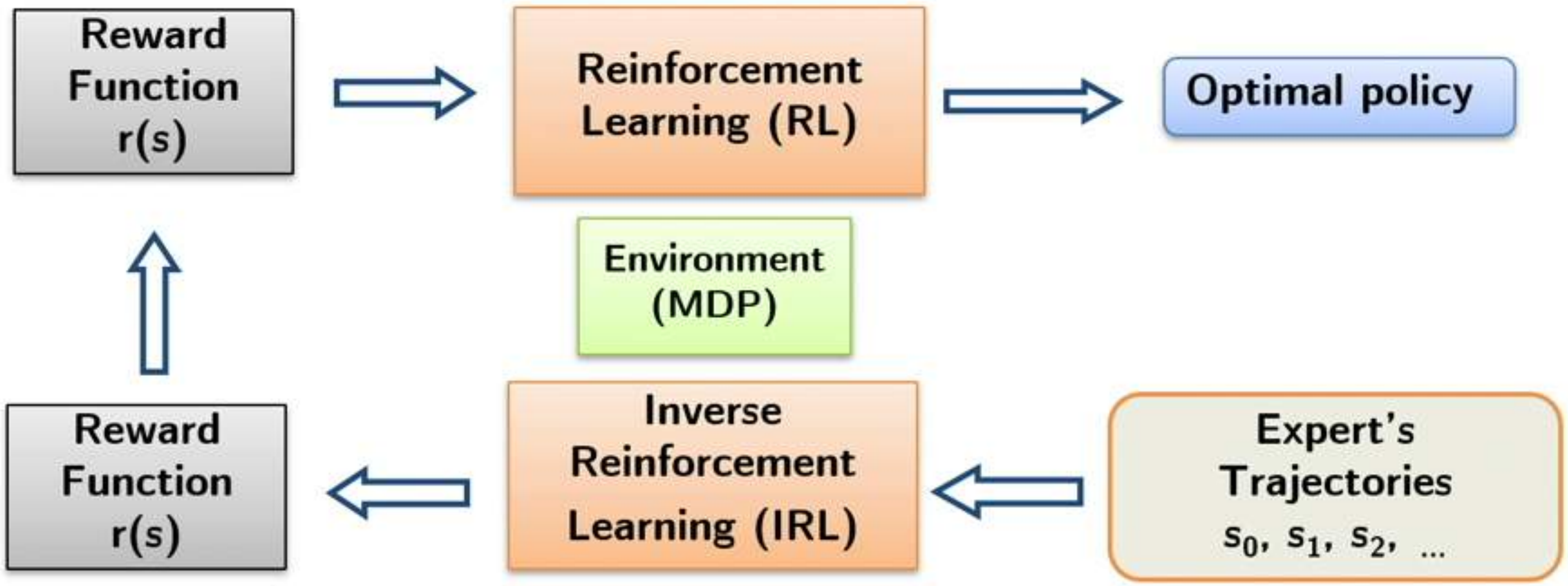
$$\mathrm{IRL}(\pi_E) = \arg\max_{r \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}} \mathbb{E}_{\pi_E}[r(s,a)] - \left(\max_{\pi \in \Pi} H(\pi) + \mathbb{E}_\pi[r(s,a)] \right)$$

↑ Expert has high reward

↓ Everything else have small reward

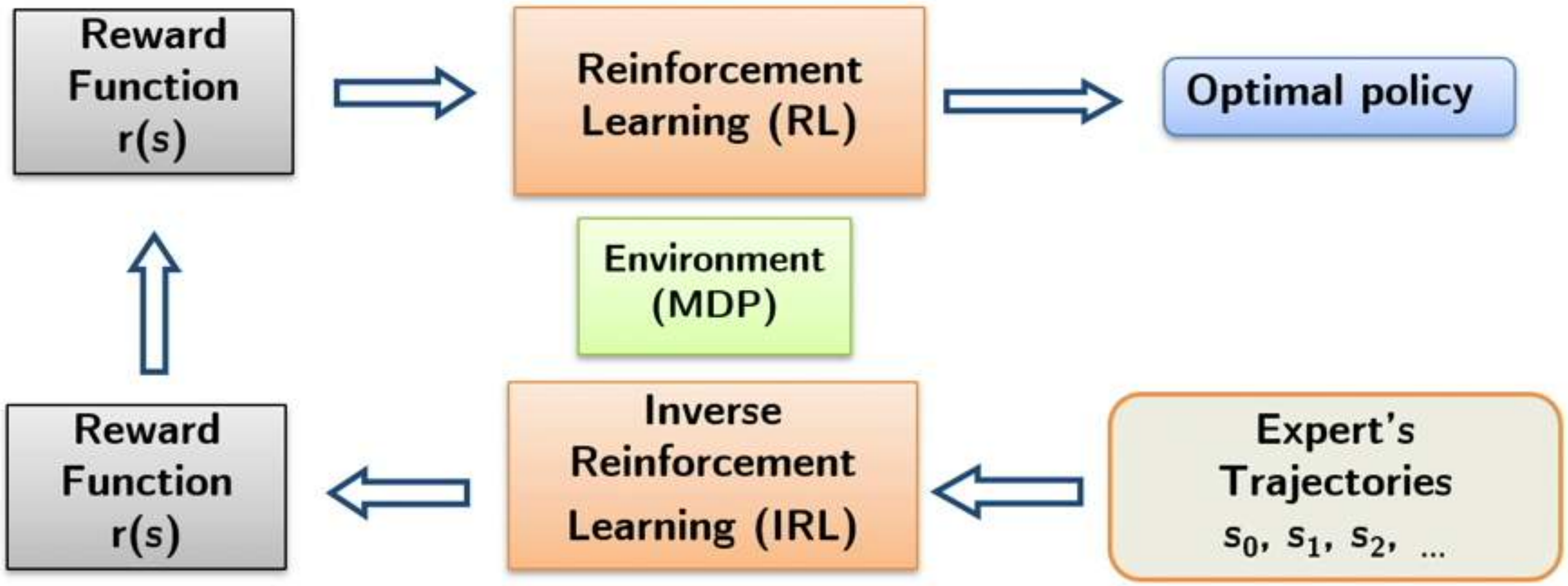
(Ziebart et al., 2010; Rust 1987)

Problem setup



$$\mathrm{IRL}_{\psi}(\pi_E) = \arg\max_{r \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}} -\psi(r) + \mathbb{E}_{\pi_E}[r(s,a)] - \left(\max_{\pi\in\Pi} H(\pi) + \mathbb{E}_{\pi}[r(s,a)]\right)$$

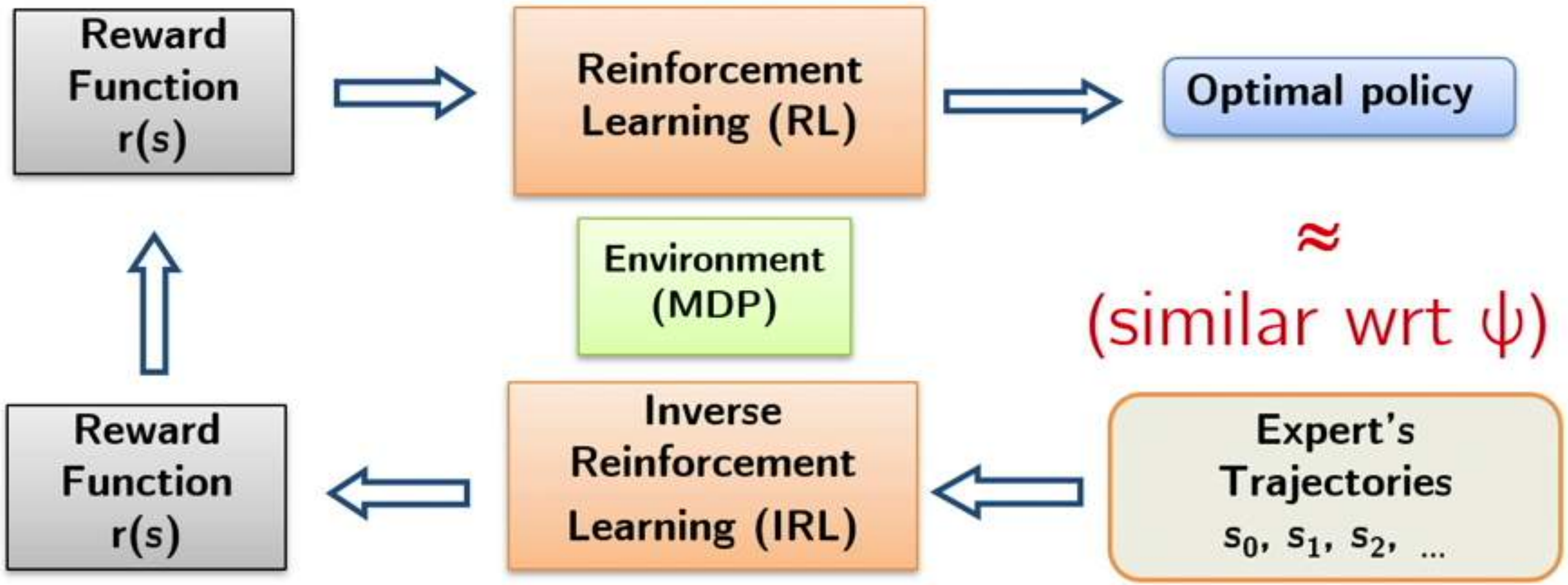
Problem setup



$$\mathrm{IRL}_{\psi}(\pi_E) = \arg\max_{r \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}} -\psi(r) + \mathbb{E}_{\pi_E}[r(s,a)] - \left(\max_{\pi\in\Pi} H(\pi) + \mathbb{E}_{\pi}[r(s,a)]\right)$$

Convex reward regularizer

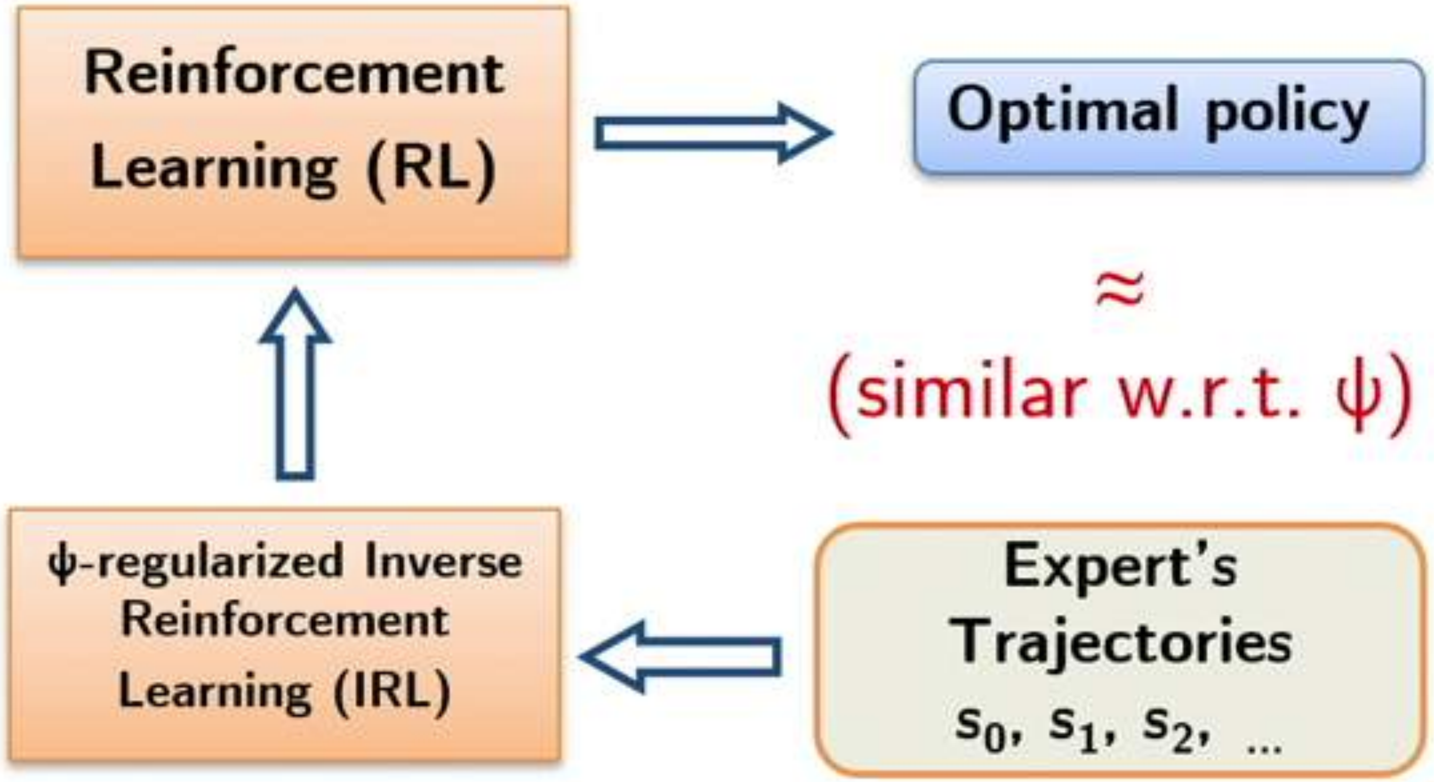
Problem setup



$$\mathrm{IRL}_{\psi}(\pi_E) = \underset{r \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}}{\arg\max} \; -\psi(r) + \mathbb{E}_{\pi_E}[r(s,a)] - \left(\max_{\pi\in\Pi} H(\pi) + \mathbb{E}_{\pi}[r(s,a)] \right)$$

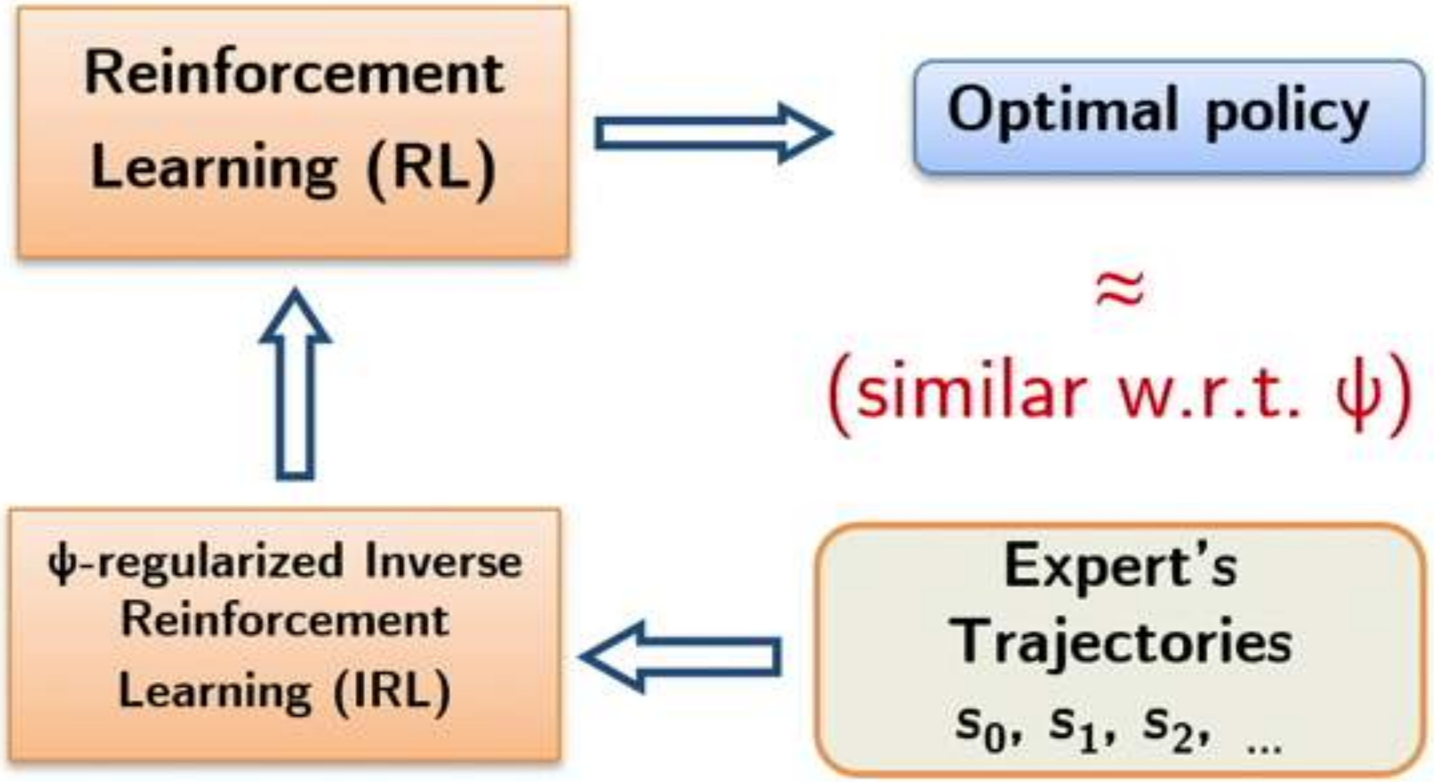
Convex reward regularizer

Combining RL and IRL



Theorem: ψ-regularized inverse reinforcement learning, implicitly, **seeks a policy whose occupancy measure is close to the expert's**, as measured by ψ^* (convex conjugate of ψ)

Combining RL and IRL



Theorem: ψ-regularized inverse reinforcement learning, implicitly, **seeks a policy whose occupancy measure is close to the expert's**, as measured by ψ^* (convex conjugate of ψ)

$$\mathrm{RL} \circ \mathrm{IRL}_{\psi}(\pi_E) = \arg\min_{\pi \in \Pi} -H(\pi) + \psi^{\star}(\rho_{\pi} - \rho_{\pi_E})$$

Takeaway

Theorem: ψ-regularized inverse reinforcement learning, implicitly, **seeks a policy whose occupancy measure is close to the expert's**, as measured by ψ^*

- Typical IRL definition: finding a reward function **r** such that the expert policy is uniquely optimal w.r.t. **r**

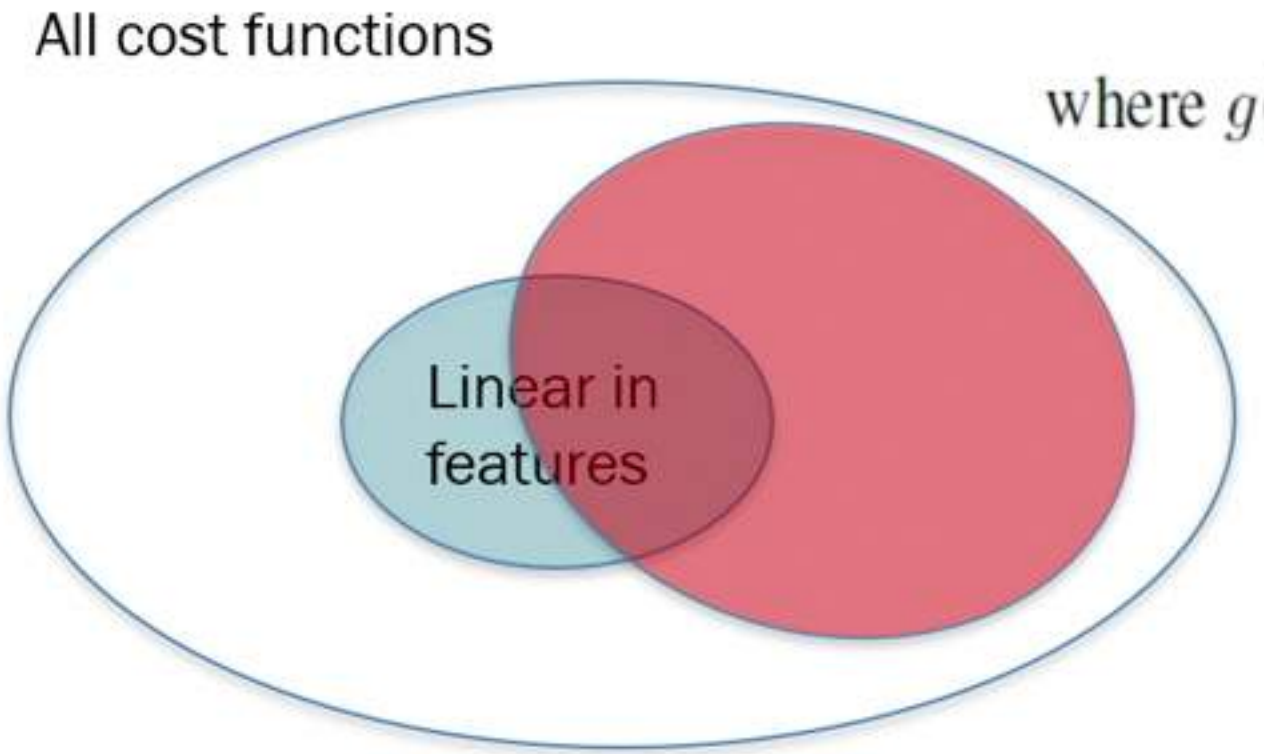
- Alternative view: IRL as a procedure that tries to induce a policy that matches the expert's occupancy measure (**generative model**)
  - Generalizes existing frameworks

Generative Adversarial Imitation Learning

- Use this regularizer

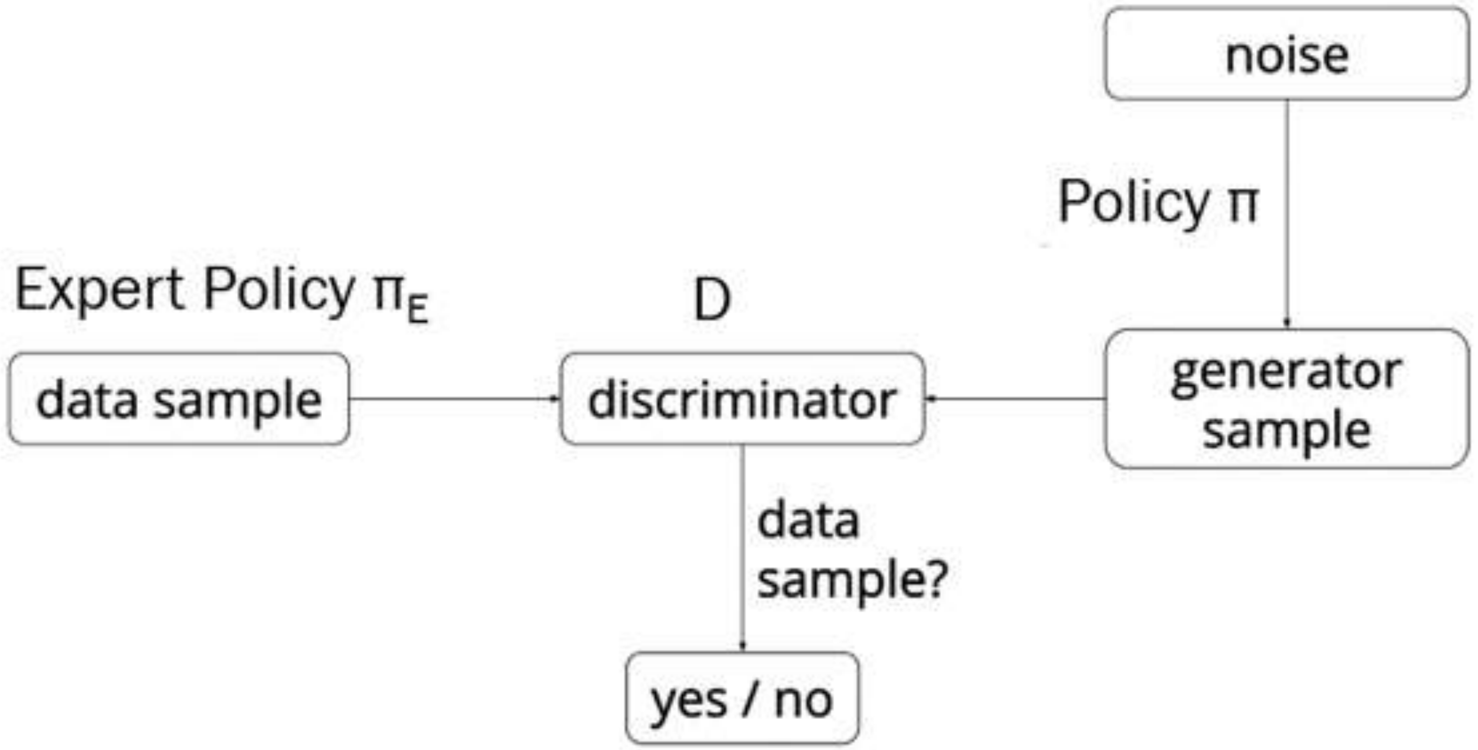
$$\psi_{\text{GA}}(c) \triangleq \begin{cases} \mathbb{E}_{\pi_E}[g(c(s,a))] & \text{if } c < 0 \\ +\infty & \text{otherwise} \end{cases}$$

$$\text{where } g(x) = \begin{cases} -x - \log(1 - e^x) & \text{if } x < 0 \\ +\infty & \text{otherwise} \end{cases}$$



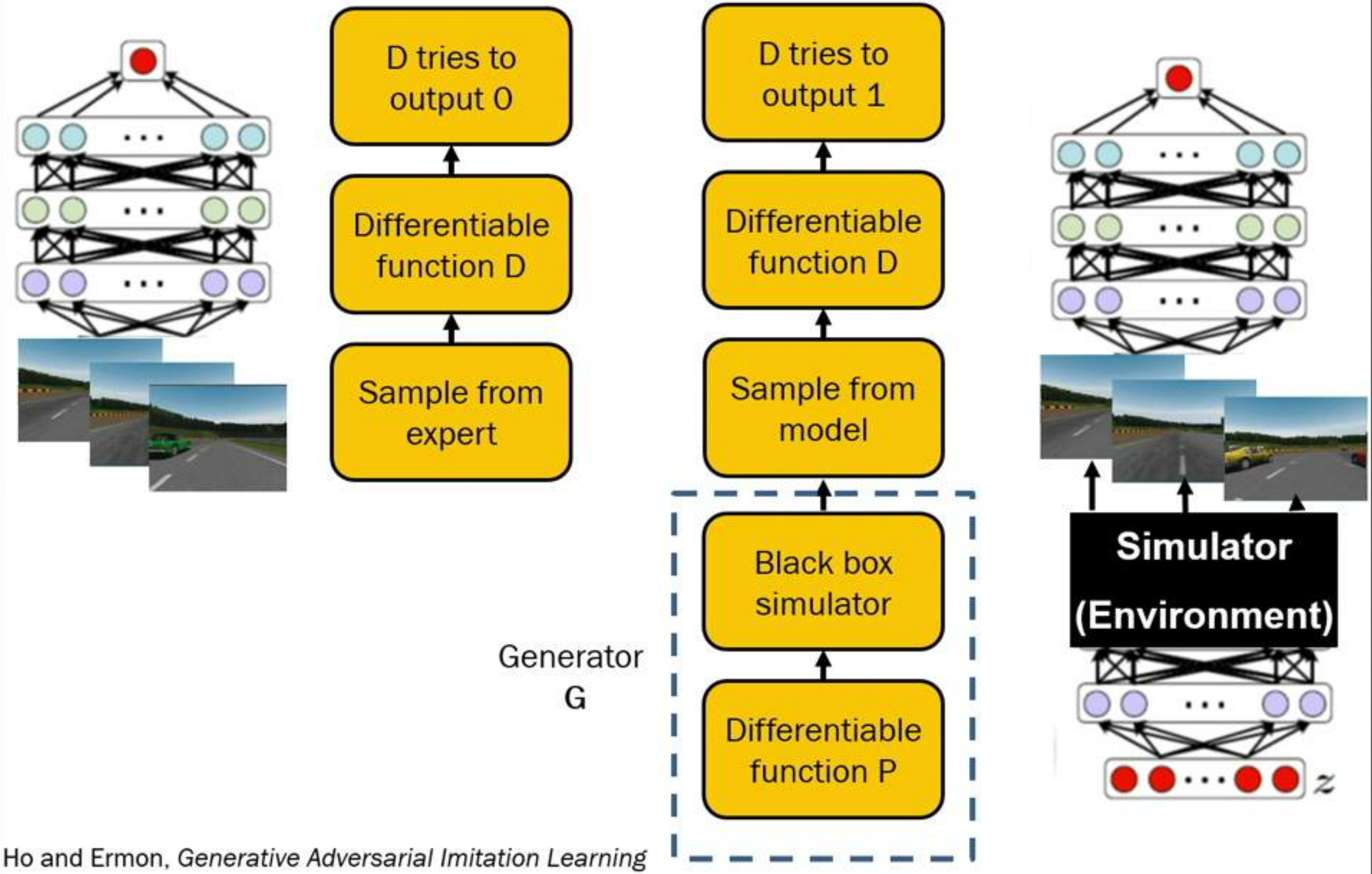
Generative Adversarial Imitation Learning

- ψ^* = optimal negative log-loss of the binary classification problem of distinguishing between state-action pairs of π and π_E



$$\psi^{\star}_{\mathrm{GA}}(\rho_\pi - \rho_{\pi_E}) = \max_{D \in (0,1)^{\mathcal{S}\times\mathcal{A}}} \mathbb{E}_{\pi_E}[\log(D(s,a))] + \mathbb{E}_{\pi}[\log(1 - D(s,a))]$$

GAIL



Ho and Ermon, *Generative Adversarial Imitation Learning*

Results

Input: driving demonstrations (TORCS Simulator)

Output policy:

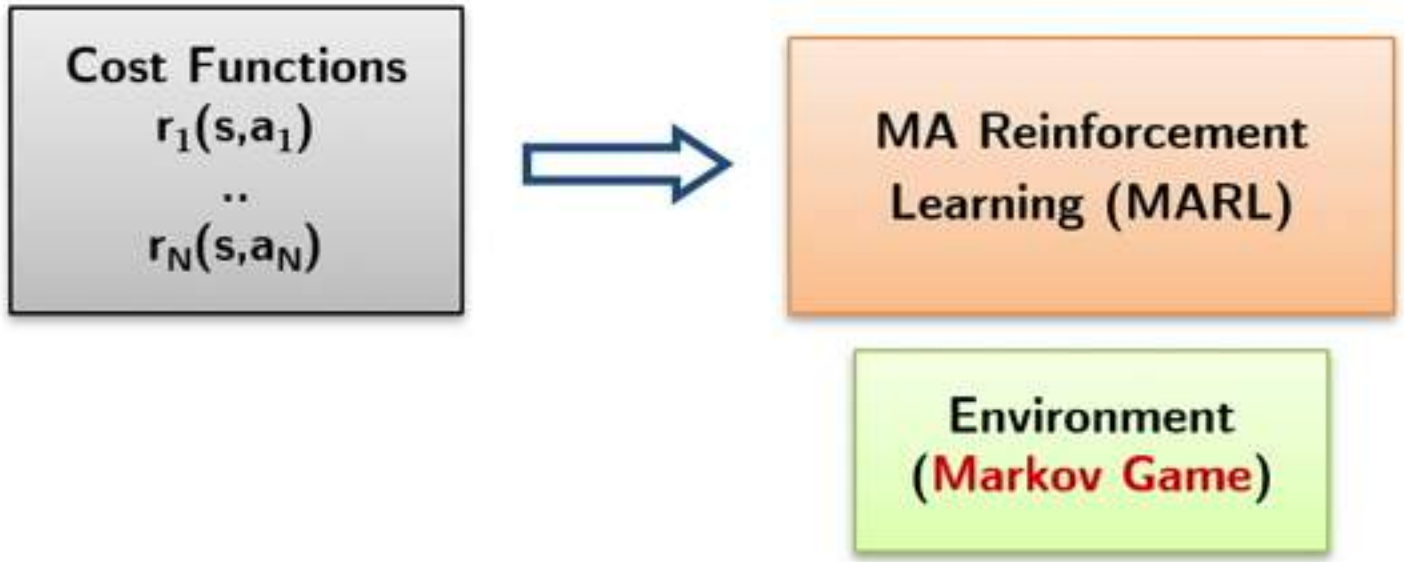
From raw visual inputs

Li et al, 2017. InfoGAIL: Interpretable Imitation Learning from Visual Demonstrations

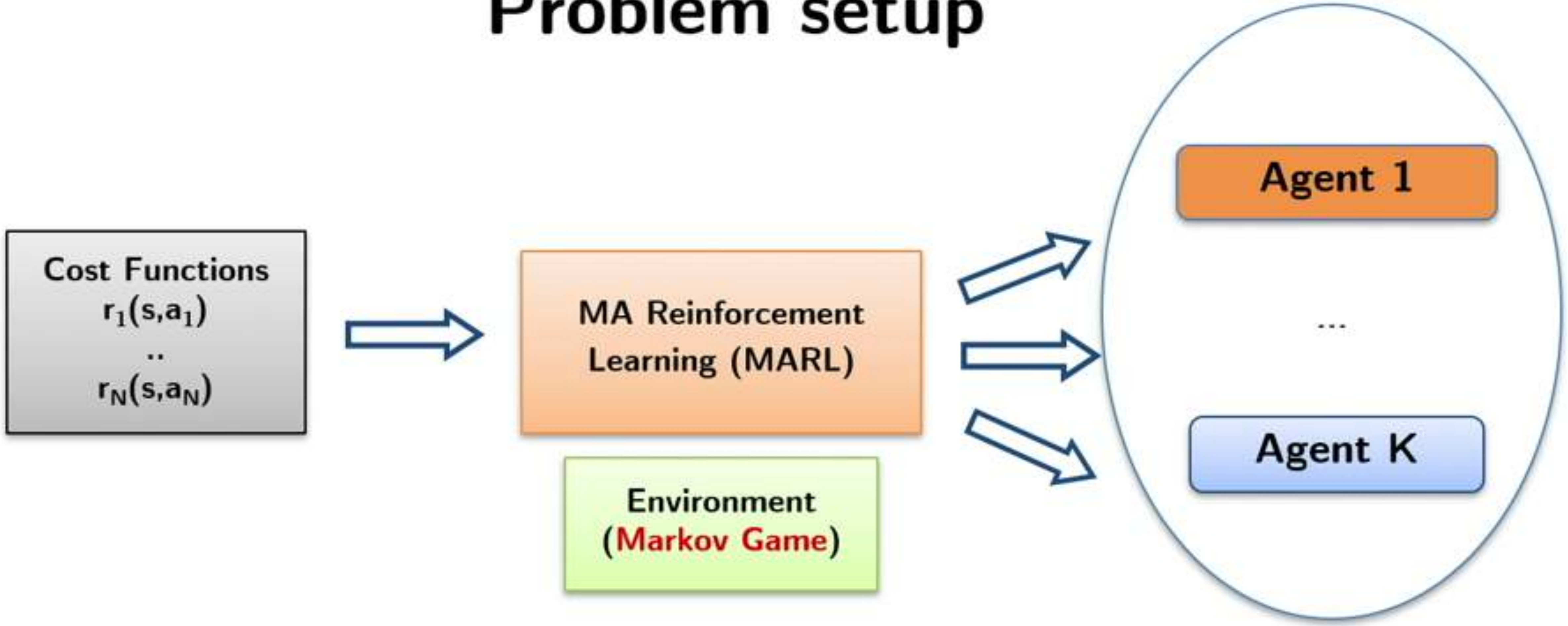
Multi-agent environments

What are the goals of these agents?

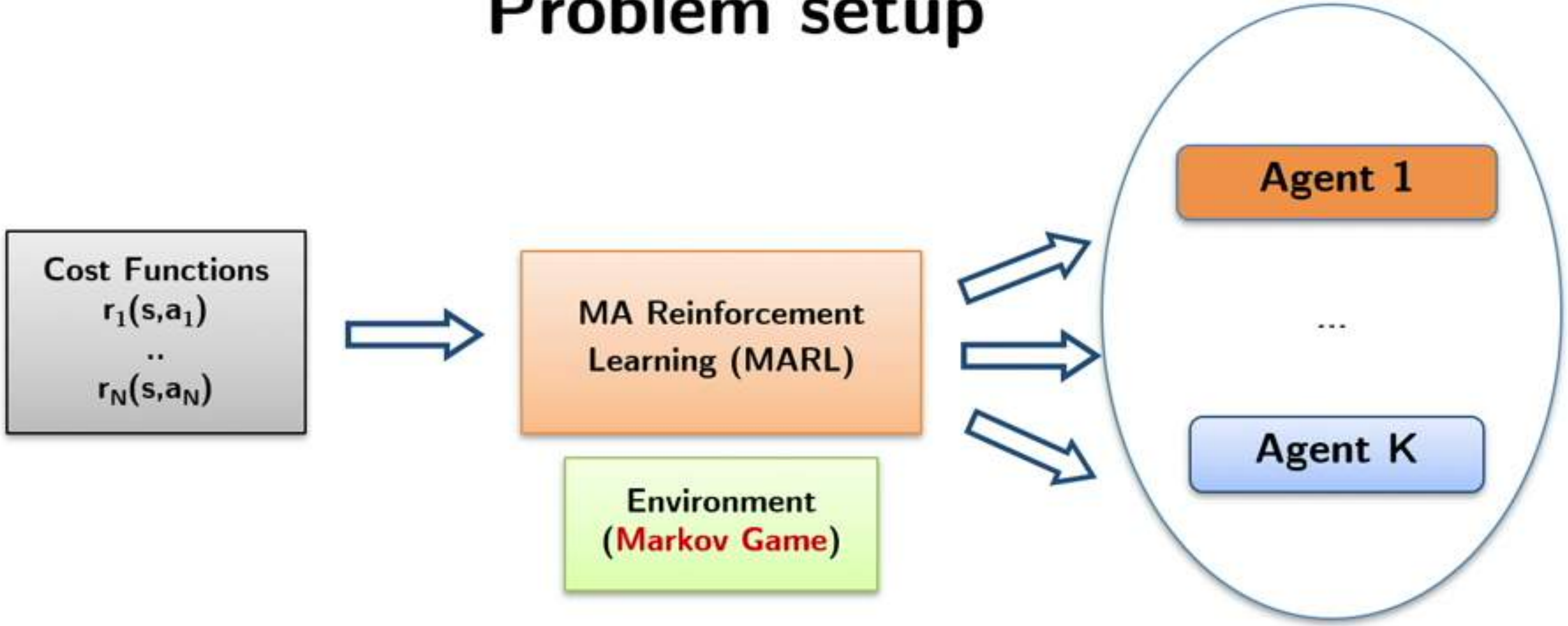
Problem setup



Problem setup

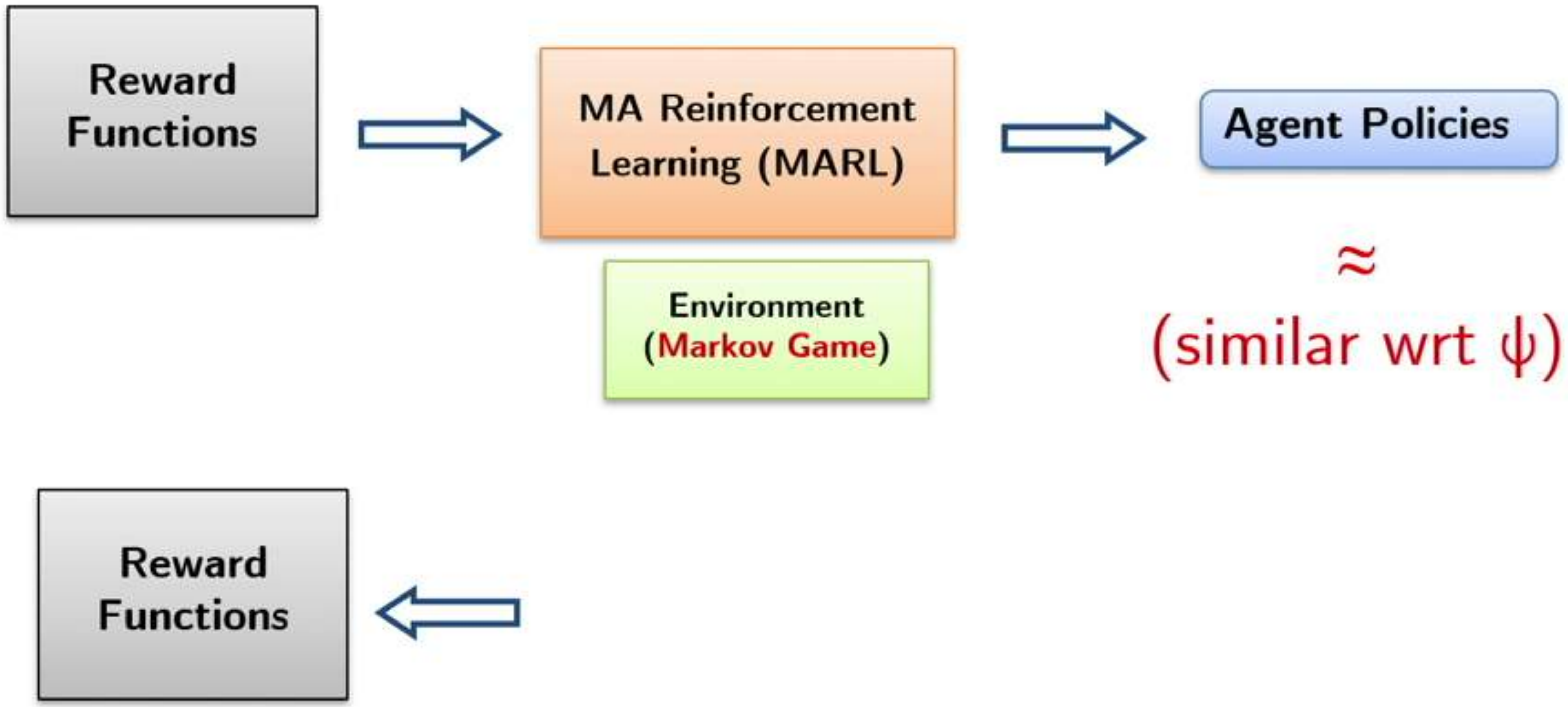


Problem setup

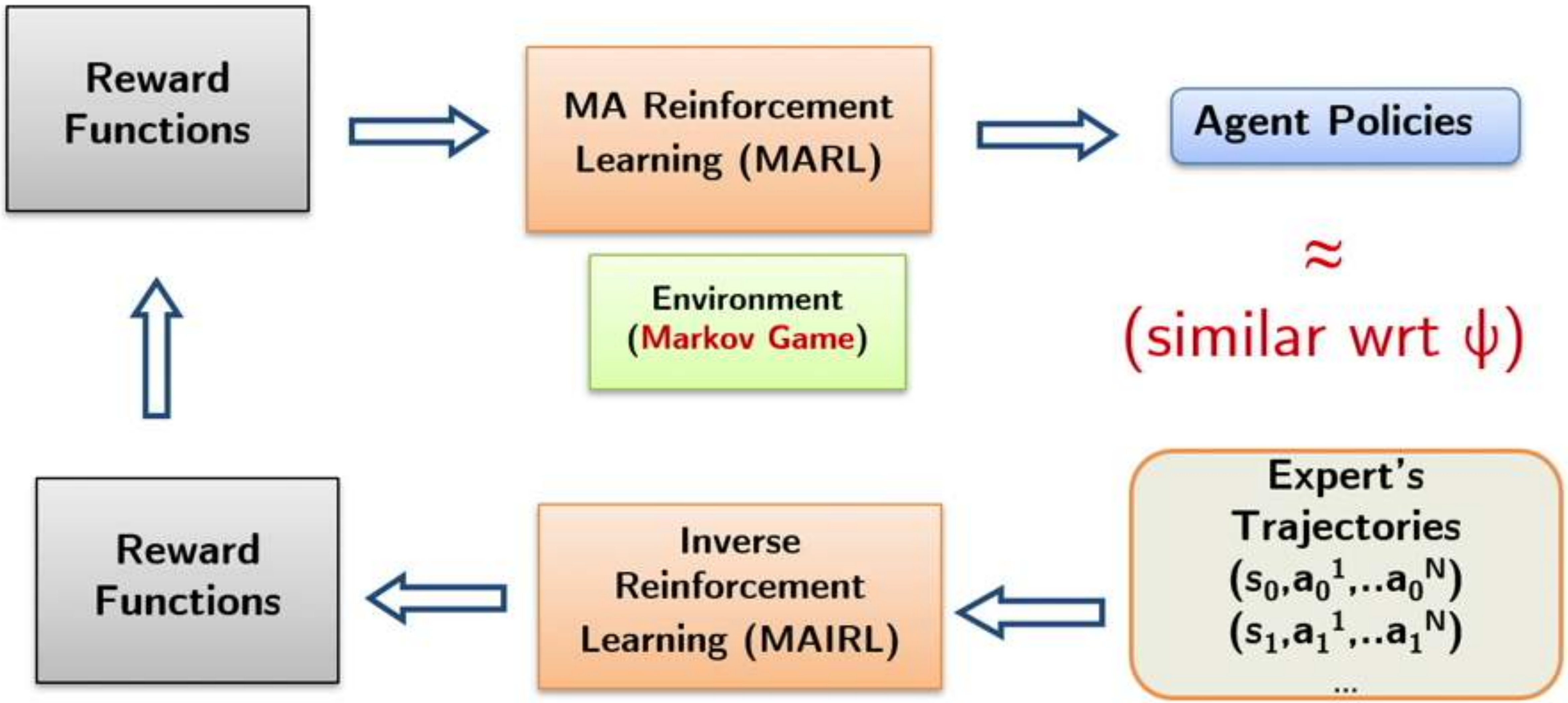


	R	L
R	0,0	10,10
L	10,10	0,0

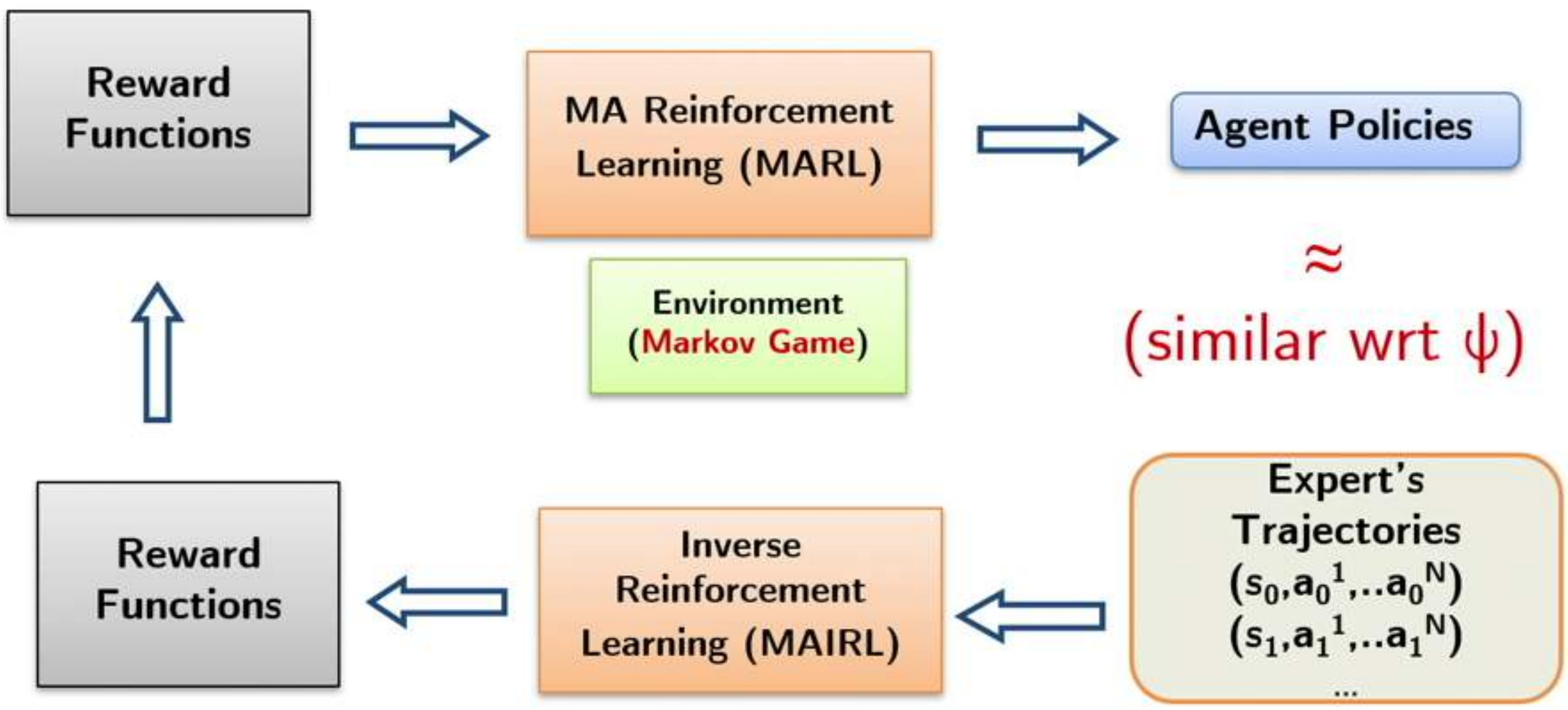
Problem setup



Problem setup



Problem setup



Can we design MAIRL that match occupancy measures?

Problem setup

For single agent IRL:

$$\mathrm{RL} \circ \mathrm{IRL}_{\psi}(\pi_E) = \arg\min_{\pi \in \Pi} -H(\pi) + \psi^{\star}(\rho_{\pi} - \rho_{\pi_E})$$

Given MARL operator, can we design MAIRL such that $\mathrm{MARL} \circ \mathrm{MAIRL}(\pi_E)$ recovers occupancy measure?

Multi-Agent Reinforcement Learning

Markov Games: extension of MDPs for multi-agents.

- Agents optimal policy depends on other agents!
- Need alternative notions of optimality

Nash Equilibrium

- No agent can achieve higher reward by unilaterally changing its policy

$$\forall i \in [1, N], \forall \hat{\pi}_i \neq \pi_i, \mathbb{E}_{\pi_i, \pi_{-i}}[r_i] \geq \mathbb{E}_{\hat{\pi}_i, \pi_{-i}}[r_i]$$

Multi-Agent Reinforcement Learning

Finding a Nash Equilibrium can be formalized into:

$$\min_{\pi \in \Pi, v \in \mathbb{R}^{\mathcal{S} \times N}} f_r(\pi, v) = \sum_{i=1}^{N} \left(\sum_{s \in \mathcal{S}} v_i(s) - \mathbb{E}_{a_i \sim \pi_i(\cdot|s)} q_i(s, a_i) \right)$$

$$\text{s.t} \quad v_i(s) \geq q_i(s, a_i) \triangleq \mathbb{E}_{\pi_{-i}} \left[r_i(s, \mathbf{a}) + \gamma \sum_{s' \in \mathcal{S}} P(s'|s, \mathbf{a}) v_i(s') \right]$$

Multi-Agent Reinforcement Learning

Finding a Nash Equilibrium can be formalized into:

"Bellman Equation"

$$\min_{\pi\in\Pi, v\in\mathbb{R}^{\mathcal{S}\times N}} f_r(\pi, v) = \sum_{i=1}^{N}\left(\sum_{s\in\mathcal{S}} v_i(s) - \mathbb{E}_{a_i\sim\pi_i(\cdot|s)} q_i(s, a_i)\right)$$

$$\text{s.t} \quad v_i(s) \geq q_i(s, a_i) \triangleq \mathbb{E}_{\pi_{-i}}\left[r_i(s, \mathbf{a}) + \gamma \sum_{s'\in\mathcal{S}} P(s'|s, \mathbf{a}) v_i(s')\right]$$

Multi-Agent Reinforcement Learning

Finding a Nash Equilibrium can be formalized into:

"Bellman Equation"

$$\min_{\pi\in\Pi, v\in\mathbb{R}^{\mathcal{S}\times N}} f_r(\pi, v) = \sum_{i=1}^{N}\left(\sum_{s\in\mathcal{S}} v_i(s) - \mathbb{E}_{a_i\sim\pi_i(\cdot|s)} q_i(s, a_i)\right)$$

$$\text{s.t} \quad v_i(s) \geq q_i(s, a_i) \triangleq \mathbb{E}_{\pi_{-i}}\left[r_i(s, \mathbf{a}) + \gamma \sum_{s'\in\mathcal{S}} P(s'|s, \mathbf{a}) v_i(s')\right]$$

Reward by deviating from current policy

Multi-Agent Reinforcement Learning

Finding a Nash Equilibrium can be formalized into:

"Bellman Equation"

$$\min_{\pi \in \Pi, v \in \mathbb{R}^{\mathcal{S} \times N}} f_r(\pi, v) = \sum_{i=1}^{N} \left(\sum_{s \in \mathcal{S}} v_i(s) - \mathbb{E}_{a_i \sim \pi_i(\cdot|s)} q_i(s, a_i) \right)$$

$$\text{s.t} \quad v_i(s) \geq q_i(s, a_i) \triangleq \mathbb{E}_{\pi_{-i}} \left[r_i(s, \mathbf{a}) + \gamma \sum_{s' \in \mathcal{S}} P(s'|s, \mathbf{a}) v_i(s') \right]$$

Reward by deviating from current policy

Objective: find value function via Value Iteration

Constraints: policy has to satisfy Nash Equilibrium

Multi-Agent Inverse Reinforcement Learning

- Assume expert is (unique) Nash under the proposed reward function.
- Expert is the (unique) global optimizer for the primal problem
- For the Lagrangian

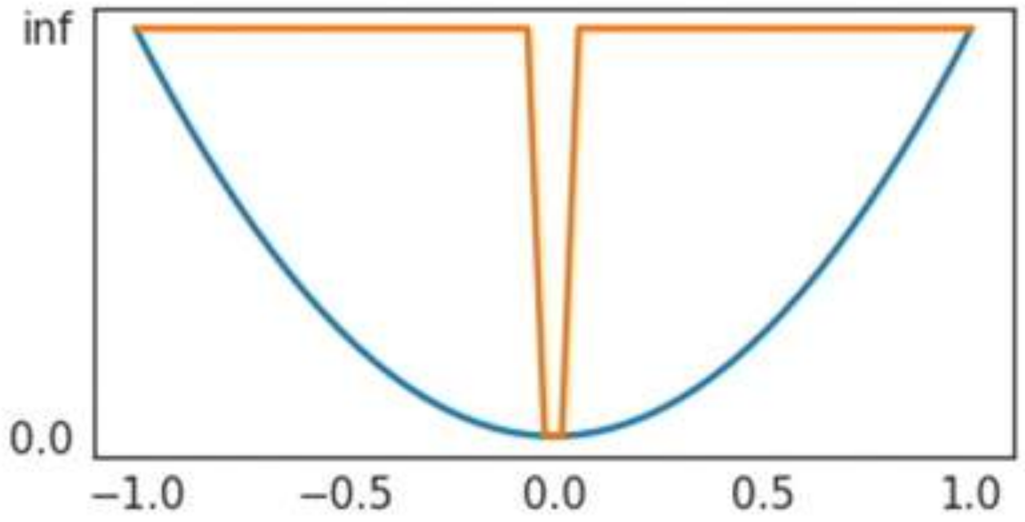
$$g(\pi) = \max_{\lambda \geq 0} f_r(\pi, v) + \sum_{i=1}^{N} \sum_{s,a_i} \lambda_{s,a_i}(q_i(s, a_i) - v_i(s))$$

Then $\quad g(\pi_E) = 0, \quad g(\pi) = \infty$

Not useful for comparing distances between policies!

Multi-Agent Inverse Reinforcement Learning

$$\mathrm{IRL}(\pi_E) = \underset{r \in \mathbb{R}^{\mathcal{S} \times \mathcal{A}}}{\arg\max}\, \mathbb{E}_{\pi_E}[r(s,a)] - \left(\max_{\pi \in \Pi} H(\pi) + \mathbb{E}_{\pi}[r(s,a)] \right)$$



- g(.) is unsuited for computing the "distance" between policies
- We construct a "smooth lower bound" of g, by choosing specific Lagrange multipliers

Step 1: Equivalent Constraints

Step 1: Equivalent Constraints

Expand the 1-step constraint to k-step constraint

Step 1: Equivalent Constraints

Expand the 1-step constraint to k-step constraint

Recall TD learning: difference with 1-step return

Step 1: Equivalent Constraints

Expand the 1-step constraint to k-step constraint

Recall TD learning: difference with 1-step return

$$r_t + \gamma \mathbb{E}_\pi [V(s_{t+1})]$$

Step 1: Equivalent Constraints

Expand the 1-step constraint to k-step constraint

Recall TD learning: difference with 1-step return

$$r_t + \gamma \mathbb{E}_\pi[V(s_{t+1})]$$

k-step TD target:

Step 1: Equivalent Constraints

Expand the 1-step constraint to k-step constraint

Recall TD learning: difference with 1-step return

$$r_t + \gamma \mathbb{E}_\pi[V(s_{t+1})]$$

k-step TD target:

$$\sum_{j=0}^{k-1} \gamma^j r_{t+j} + \gamma^k \mathbb{E}_\pi[V(s_{t+k})]$$

Step 1: Equivalent Constraints

Define $Q_i^{(k)}(\{s^j, a_i^j\}_{j=0}^{k-1}, a_i^k)$ as discounted return

- For agent i
- When agent i took and visited $\{s^j, a_i^j\}_{j=0}^{k-1}$
- And other agents act according to their policies

Change constraints to

$$v_i(s^{(0)}) \geq Q_i^{(k)}(\{s^{(j)}, a_i^{(j)}\}_{j=0}^{k-1}, a_i^{(k)})$$

Still ensures Nash Equilibrium!

Step 2: Find the Lagrange Multipliers

Step 2: Find the Lagrange Multipliers

Consider the Lagrangian

$$\max_{\lambda \geq 0} \min_{\pi} L_{\mathbf{r}}^{(t+1)}(\pi, \lambda) \triangleq \sum_{i=1}^{N} \sum_{\tau_i \in \mathcal{T}_i^t} \lambda(\tau_i) \left(Q_i^{(t)}(\tau_i; \pi, \mathbf{r}) - v_i(s^{(0)}; \pi, \mathbf{r}) \right)$$

Step 2: Find the Lagrange Multipliers

Consider the Lagrangian

$$\max_{\lambda \geq 0} \min_{\pi} L_{\mathbf{r}}^{(t+1)}(\pi, \lambda) \triangleq \sum_{i=1}^{N} \sum_{\tau_i \in \mathcal{T}_i^t} \lambda(\tau_i) \left(Q_i^{(t)}(\tau_i; \pi, \mathbf{r}) - v_i(s^{(0)}; \pi, \mathbf{r}) \right)$$

Length t trajectories

Step 2: Find the Lagrange Multipliers

Consider the Lagrangian

$$\max_{\lambda \geq 0} \min_{\pi} L_{\mathbf{r}}^{(t+1)}(\pi, \lambda) \triangleq \sum_{i=1}^{N} \sum_{\tau_i \in \mathcal{T}_i^t} \lambda(\tau_i) \left(Q_i^{(t)}(\tau_i; \pi, \mathbf{r}) - v_i(s^{(0)}; \pi, \mathbf{r}) \right)$$

Length t trajectories

For any two policies $\pi^\star$ and π,

Let $\lambda_\pi^\star(\tau_i)$ be the probability of generating the sequence with $(\pi_i, \pi_{-i}^\star)$, then

Step 2: Find the Lagrange Multipliers

Consider the Lagrangian

$$\max_{\lambda \geq 0} \min_{\pi} L_{\mathbf{r}}^{(t+1)}(\pi, \lambda) \triangleq \sum_{i=1}^{N} \sum_{\tau_i \in \mathcal{T}_i^t} \lambda(\tau_i) \left(Q_i^{(t)}(\tau_i; \pi, \mathbf{r}) - v_i(s^{(0)}; \pi, \mathbf{r}) \right)$$

Length t trajectories

For any two policies $\pi^\star$ and π,

Let $\lambda_\pi^\star(\tau_i)$ be the probability of generating the sequence with $(\pi_i, \pi_{-i}^\star)$, then

$$\lim_{t \to \infty} L_r^{(t+1)}(\pi^\star, \lambda_\pi^\star) = \sum_{i=1}^{N} \left(\mathbb{E}_{\pi_i, \pi_{-i}^\star}[r_i(s,a)] - \mathbb{E}_{\pi_i^\star, \pi_{-i}^\star}[r_i(s,a)] \right)$$

MAIRL Operator

This motivates the following MAIRL operator

$$\arg\max_{\mathbf{r}} -\psi(\mathbf{r}) + \sum_{i=1}^{N}(\mathbb{E}_{\pi_E}[r_i]) - \left(\max_{\pi} \sum_{i=1}^{N}(\beta H_i(\pi_i) + \mathbb{E}_{\pi_i,\pi_{E_{-i}}}[r_i])\right)$$

Strictly generalizes the IRL operator when N=1!

MAIRL Operator

This motivates the following MAIRL operator

$$\arg\max_{\mathbf{r}} -\psi(\mathbf{r}) + \sum_{i=1}^{N}(\mathbb{E}_{\pi_E}[r_i]) - \left(\max_{\pi}\sum_{i=1}^{N}(\beta H_i(\pi_i) + \mathbb{E}_{\pi_i,\pi_{E_{-i}}}[r_i])\right)$$

↑ Expert has high reward

Strictly generalizes the IRL operator when N=1!

MAIRL Operator

This motivates the following MAIRL operator

$$\arg\max_{\mathbf{r}} -\psi(\mathbf{r}) + \sum_{i=1}^{N}(\mathbb{E}_{\pi_E}[r_i]) - \left(\max_{\pi}\sum_{i=1}^{N}(\beta H_i(\pi_i) + \mathbb{E}_{\pi_i,\pi_{E_{-i}}}[r_i])\right)$$

↑ Expert has high reward

↓ Everything else have small reward

Strictly generalizes the IRL operator when N=1!

Multi-Agent Imitation Learning

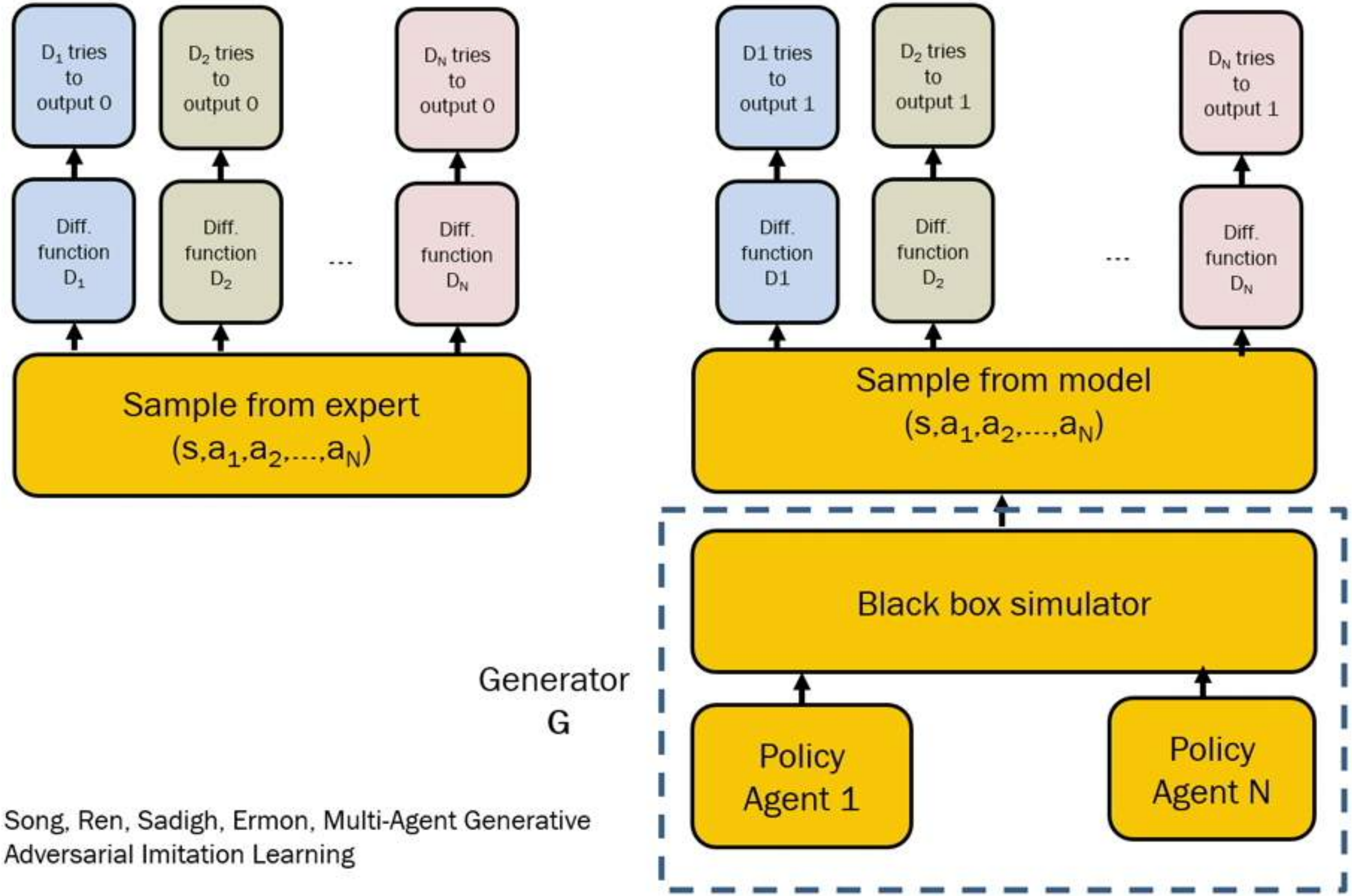
Assume that reward function is additively separable

$$\psi(\mathbf{r}) = \sum_{i=1}^{N} \psi_i(r_i)$$

Then

$$\mathrm{MARL} \circ \mathrm{MAIRL}_{\psi}(\pi_E) = \arg\min_{\pi \in \Pi} \sum_{i=1}^{N} -\beta H_i(\pi_i) + \psi_i^{\star}(\rho_{\pi_{i,E_{-i}}} - \rho_{\pi_E})$$

MAGAIL



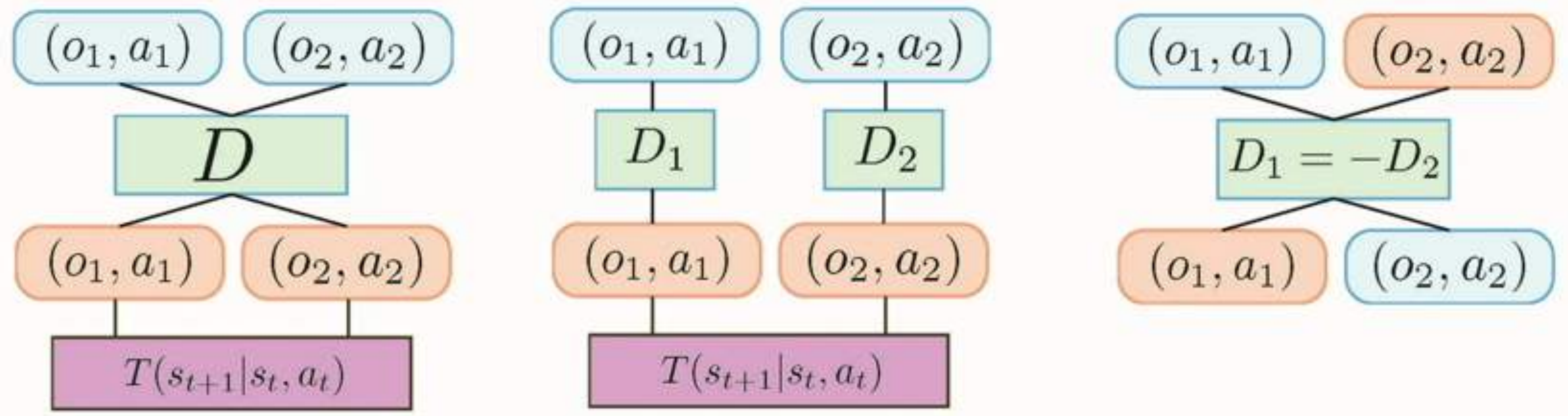
Song, Ren, Sadigh, Ermon, Multi-Agent Generative Adversarial Imitation Learning

MAGAIL

Incorporate knowledge on reward structure via regularizers

MAGAIL

Incorporate knowledge on reward structure via regularizers

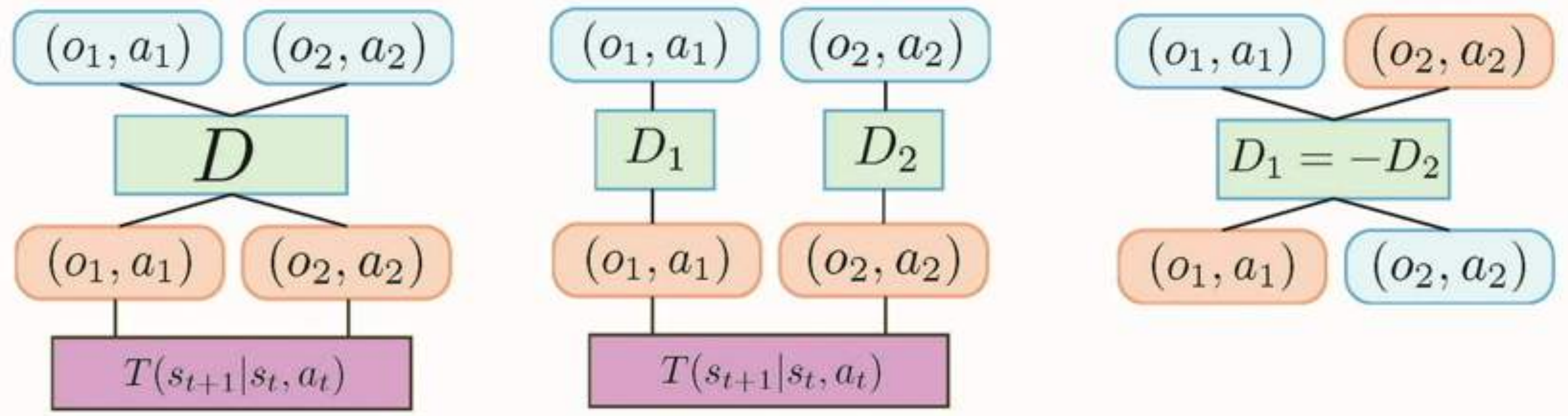


Centralized Decentralized Zero-sum

MAGAIL

Incorporate knowledge on reward structure via regularizers

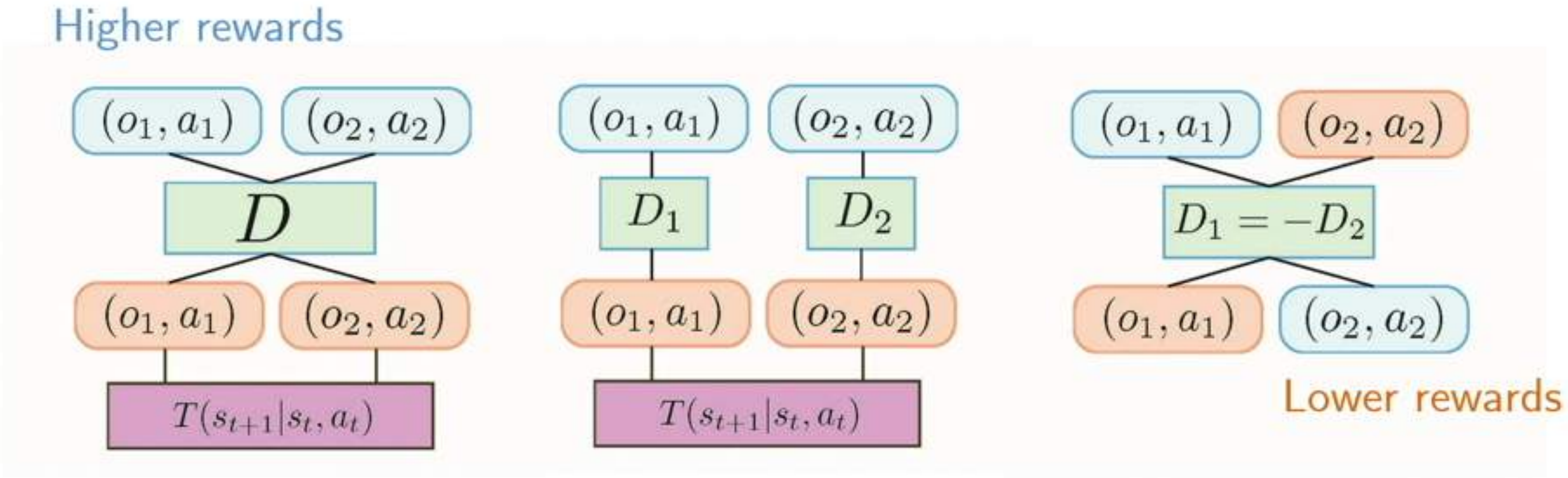
Higher rewards



Centralized   Decentralized   Zero-sum

MAGAIL

Incorporate knowledge on reward structure via regularizers



Centralized Decentralized Zero-sum

Experiments

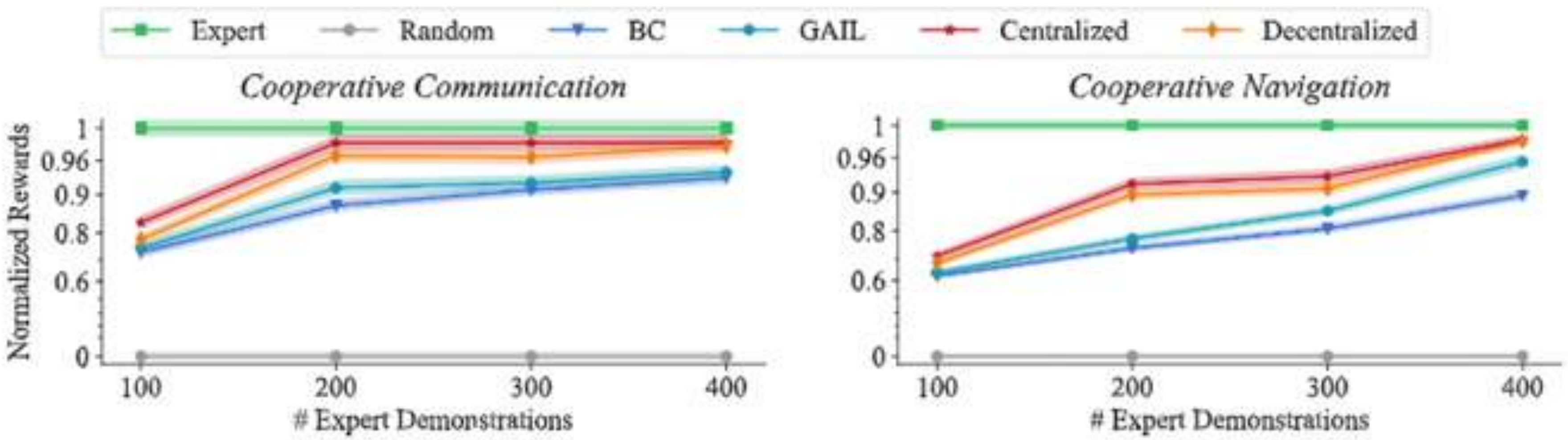


Table 1: Average agent rewards in competitive tasks. We compare behavior cloning (BC), GAIL (G), Centralized (C), Decentralized (D), and Zero-Sum (ZS) methods. Best marked in bold (high vs. low rewards is preferable depending on the agent vs. adversary role).

Task	Predator-Prey								
Agent	Behavior Cloning					G	C	D	ZS
Adversary	BC	G	C	D	ZS	Behavior Cloning			
Rewards	-93.20	-93.71	-93.75	-95.22	**-95.48**	-90.55	-91.36	**-85.00**	-89.4
Task	Keep-Away								
Agent	Behavior Cloning					G	C	D	ZS
Adversary	BC	G	C	D	ZS	Behavior Cloning			
Rewards	24.22	24.04	23.28	23.56	**23.19**	26.22	26.61	**28.73**	27.80

Suboptimal demos

Expert

Conclusions

Conclusions

- IRL is a dual of an occupancy measure matching problem (generative modeling)

Conclusions

- IRL is a dual of an occupancy measure matching problem (generative modeling)
- Multi-agent cases are complicated by alternative optimality notions (e.g. Nash Equilibrium)

Conclusions

- IRL is a dual of an occupancy measure matching problem (generative modeling)
- Multi-agent cases are complicated by alternative optimality notions (e.g. Nash Equilibrium)
- Under certain cases, we can link multi-agent imitation learning to occupancy measure matching

Conclusions

- IRL is a dual of an occupancy measure matching problem (generative modeling)
- Multi-agent cases are complicated by alternative optimality notions (e.g. Nash Equilibrium)
- Under certain cases, we can link multi-agent imitation learning to occupancy measure matching
- Limitations exist (e.g. zero-sum)

Learning Fair and Controllable Representations

Jiaming Song

w/ Ria Kalluri, Aditya Grover, Shengjia Zhao, Stefano Ermon

Stanford University

Problem setup

- X: features of an individual

Problem setup

- X: features of an individual
- U: sensitive attribute (e.g. gender)

Problem setup

- X: features of an individual
- U: sensitive attribute (e.g. gender)
- $C = c(X, U)$ predicted values

Problem setup

- X: features of an individual
- U: sensitive attribute (e.g. gender)
- C = c(X, U) predicted values
- Y: target variable (labels)

Problem setup

- X: features of an individual
- U: sensitive attribute (e.g. gender)
- $C = c(X, U)$ predicted values
- Y: target variable (labels)
- Z: representation (used for downstream tasks)

Problem setup

- X: features of an individual
- U: sensitive attribute (e.g. gender)
- $C = c(X, U)$ predicted values
- Y: target variable (labels)
- Z: representation (used for downstream tasks)

Make accurate predictions while protecting U.

Problem setup

- X: features of an individual
- U: sensitive attribute (e.g. gender)
- $C = c(X, U)$ predicted values
- Y: target variable (labels)
- Z: representation (used for downstream tasks)

Make accurate predictions while protecting U.

- "Give loan according to credit but fair to race"

Examples of Fairness Notions

Examples of Fairness Notions

- Demographic parity
  - C and U are independent.
  - $I(C; U) = 0$

Examples of Fairness Notions

- Demographic parity
  - C and U are independent.
  - $I(C; U) = 0$
- Equalized odds
  - C and U are independent conditional on Y
  - $I(C; U|Y) = 0$

Examples of Fairness Notions

- Demographic parity
  - C and U are independent.
  - $I(C; U) = 0$
- Equalized odds
  - C and U are independent conditional on Y
  - $I(C; U|Y) = 0$
- Equalized opportunity
  - C and U are independent conditional on $Y = 1$
  - $I(C; U|Y{=}1) = 0$

Examples of Fairness Notions

- Demographic parity
  - C and U are independent.
  - $I(C; U) = 0$
- Equalized odds
  - C and U are independent conditional on Y
  - $I(C; U|Y) = 0$
- Equalized opportunity
  - C and U are independent conditional on $Y = 1$
  - $I(C; U|Y{=}1) = 0$

Not all notions can be satisfied at once!

Representation Learning

Representation Learning

- Learn a representation Z, and use Z to predict Y

Representation Learning

- Learn a representation Z, and use Z to predict Y
- "Data Preprocessing"

Learning Fair and Expressive Representations

$\max I(X; Z|U) \quad \min I(U; Z)$

Optimization Problem

Optimization Problem

- Maximize “expressiveness”

Optimization Problem

- Maximize “expressiveness”
- Under “fairness” constraints

Optimization Problem

- Maximize "expressiveness"
- Under "fairness" constraints
- Prediction model

Optimization Problem

- Maximize “expressiveness”
- Under “fairness” constraints
- Prediction model $q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{u})$

Optimization Problem

- Maximize “expressiveness”
- Under “fairness” constraints
- Prediction model $q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{u})$
- Data distribution

Optimization Problem

- Maximize “expressiveness”
- Under “fairness” constraints
- Prediction model $q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{u})$
- Data distribution $q(\mathbf{x}, \mathbf{u})$

Optimization Problem

- Maximize "expressiveness"
- Under "fairness" constraints
- Prediction model $q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{u})$
- Data distribution $q(\mathbf{x}, \mathbf{u})$
- Constrained optimization problem
  - e.g. demographic parity

Optimization Problem

- Maximize "expressiveness"
- Under "fairness" constraints
- Prediction model $q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{u})$
- Data distribution $q(\mathbf{x}, \mathbf{u})$
- Constrained optimization problem
  - e.g. demographic parity

$$\max I_q(\mathbf{x}; \mathbf{z}|\mathbf{u})$$
$$\text{s.t.} \quad I_q(\mathbf{z}; \mathbf{u}) < \epsilon$$

Optimization Problem

- Maximize "expressiveness"
- Under "fairness" constraints
- Prediction model $q_\phi(\mathbf{z}|\mathbf{x},\mathbf{u})$
- Data distribution $q(\mathbf{x},\mathbf{u})$
- Constrained optimization problem
  - e.g. demographic parity

$$\max I_q(\mathbf{x};\mathbf{z}|\mathbf{u})$$
$$\text{s.t.} \quad I_q(\mathbf{z};\mathbf{u}) < \epsilon$$

Quantity determined by user (hyperparameter)

Tractable Bounds for Mutual Information

Tractable Bounds for Mutual Information

These quantities are not “tractable”!

Tractable Bounds for Mutual Information

These quantities are not “tractable”!

$$I_q(\mathbf{x};\mathbf{z}|\mathbf{u}) = \mathbb{E}_{q_\phi(\mathbf{x},\mathbf{z},\mathbf{u})}[\log q_\phi(\mathbf{x},\mathbf{z}|\mathbf{u}) - \log q(\mathbf{x}|\mathbf{u}) - \log q_\phi(\mathbf{z}|\mathbf{u})]$$

Tractable Bounds for Mutual Information

These quantities are not "tractable"!

$$I_q(\mathbf{x};\mathbf{z}|\mathbf{u}) = \mathbb{E}_{q_\phi(\mathbf{x},\mathbf{z},\mathbf{u})}[\log q_\phi(\mathbf{x},\mathbf{z}|\mathbf{u}) - \log q(\mathbf{x}|\mathbf{u}) - \log q_\phi(\mathbf{z}|\mathbf{u})]$$

$$I_q(\mathbf{z};\mathbf{u}) = \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{u})}[\log q_\phi(\mathbf{z}|\mathbf{u}) - \log q_\phi(\mathbf{z})]$$

Tractable Bounds for Mutual Information

These quantities are not "tractable"!

$$I_q(\mathbf{x};\mathbf{z}|\mathbf{u}) = \mathbb{E}_{q_\phi(\mathbf{x},\mathbf{z},\mathbf{u})}[\log q_\phi(\mathbf{x},\mathbf{z}|\mathbf{u}) - \log q(\mathbf{x}|\mathbf{u}) - \log q_\phi(\mathbf{z}|\mathbf{u})]$$

$$I_q(\mathbf{z};\mathbf{u}) = \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{u})}[\log q_\phi(\mathbf{z}|\mathbf{u}) - \log q_\phi(\mathbf{z})]$$

Only $q_\phi(\mathbf{z}|\mathbf{x},\mathbf{u})$ has tractable log density!

$$\max I_q(\mathbf{x}; \mathbf{z}|\mathbf{u})$$

- Introduce a parametrized distribution

Fixing a Broken ELBO, Alemi et al. ICML 2018.

$$\max I_q(\mathbf{x};\mathbf{z}|\mathbf{u})$$

- Introduce a parametrized distribution $p_\theta(\mathbf{x}|\mathbf{z},\mathbf{u})$

Fixing a Broken ELBO, Alemi et al. ICML 2018.

$$\max I_q(\mathbf{x};\mathbf{z}|\mathbf{u})$$

- Introduce a parametrized distribution $p_\theta(\mathbf{x}|\mathbf{z},\mathbf{u})$

$$I_q(\mathbf{x};\mathbf{z}|\mathbf{u}) = \mathbb{E}_{q_\phi(\mathbf{x},\mathbf{z},\mathbf{u})}[\log p_\theta(\mathbf{x}|\mathbf{z},\mathbf{u})] + H_q(\mathbf{x}|\mathbf{u}) + \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{u})} D_{\mathrm{KL}}(q_\phi(\mathbf{x}|\mathbf{z},\mathbf{u}) \| p_\theta(\mathbf{x}|\mathbf{z},\mathbf{u}))$$

Fixing a Broken ELBO, Alemi et al. ICML 2018.

$$\max I_q(\mathbf{x}; \mathbf{z}|\mathbf{u})$$

- Introduce a parametrized distribution $p_\theta(\mathbf{x}|\mathbf{z}, \mathbf{u})$

$$I_q(\mathbf{x}; \mathbf{z}|\mathbf{u}) = \mathbb{E}_{q_\phi(\mathbf{x},\mathbf{z},\mathbf{u})}[\log p_\theta(\mathbf{x}|\mathbf{z}, \mathbf{u})] + \boxed{H_q(\mathbf{x}|\mathbf{u})} + \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{u})} D_{\mathrm{KL}}(q_\phi(\mathbf{x}|\mathbf{z}, \mathbf{u}) \| p_\theta(\mathbf{x}|\mathbf{z}, \mathbf{u}))$$

"Constant Entropy"

Fixing a Broken ELBO, Alemi et al. ICML 2018.

$$\max I_q(\mathbf{x};\mathbf{z}|\mathbf{u})$$

- Introduce a parametrized distribution $p_\theta(\mathbf{x}|\mathbf{z},\mathbf{u})$

$$I_q(\mathbf{x};\mathbf{z}|\mathbf{u}) = \mathbb{E}_{q_\phi(\mathbf{x},\mathbf{z},\mathbf{u})}[\log p_\theta(\mathbf{x}|\mathbf{z},\mathbf{u})] + H_q(\mathbf{x}|\mathbf{u}) + \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{u})} D_{\mathrm{KL}}(q_\phi(\mathbf{x}|\mathbf{z},\mathbf{u}) \| p_\theta(\mathbf{x}|\mathbf{z},\mathbf{u}))$$

"Constant Entropy"

KL divergence > 0

Fixing a Broken ELBO, Alemi et al. ICML 2018.

$$\max I_q(\mathbf{x}; \mathbf{z}|\mathbf{u})$$

- Introduce a parametrized distribution $p_\theta(\mathbf{x}|\mathbf{z}, \mathbf{u})$

$$I_q(\mathbf{x}; \mathbf{z}|\mathbf{u}) = \mathbb{E}_{q_\phi(\mathbf{x},\mathbf{z},\mathbf{u})}[\log p_\theta(\mathbf{x}|\mathbf{z}, \mathbf{u})] + H_q(\mathbf{x}|\mathbf{u}) + \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{u})} D_{\mathrm{KL}}(q_\phi(\mathbf{x}|\mathbf{z}, \mathbf{u}) \| p_\theta(\mathbf{x}|\mathbf{z}, \mathbf{u}))$$

"Reconstruction Error" "Distortion" — "Constant Entropy" — KL divergence > 0

Fixing a Broken ELBO, Alemi et al. ICML 2018.

$$\max I_q(\mathbf{x};\mathbf{z}|\mathbf{u})$$

- Introduce a parametrized distribution $p_\theta(\mathbf{x}|\mathbf{z},\mathbf{u})$

$$I_q(\mathbf{x};\mathbf{z}|\mathbf{u}) = \mathbb{E}_{q_\phi(\mathbf{x},\mathbf{z},\mathbf{u})}[\log p_\theta(\mathbf{x}|\mathbf{z},\mathbf{u})] + H_q(\mathbf{x}|\mathbf{u}) + \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{u})} D_{\mathrm{KL}}(q_\phi(\mathbf{x}|\mathbf{z},\mathbf{u}) \| p_\theta(\mathbf{x}|\mathbf{z},\mathbf{u}))$$

"Reconstruction Error"
"Distortion"

"Constant
Entropy"

KL divergence > 0

Use "Reconstruction Error" as lower bound (up to constant)!

Fixing a Broken ELBO, Alemi et al. ICML 2018.

$$\min I_q(\mathbf{u}, \mathbf{z})$$

A Lagrangian Perspective of Latent Variable Generative Models, Zhao, Song, Ermon. UAI 2018.

$$\min I_q(\mathbf{u}, \mathbf{z})$$

$$I_q(\mathbf{z}; \mathbf{u}) \leq I_q(\mathbf{z}; \mathbf{x}, \mathbf{u}) = \mathbb{E}_{q(\mathbf{x},\mathbf{u})} D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{u}) \| p(\mathbf{z})) - D_{\mathrm{KL}}(q_\phi(\mathbf{z}) \| p(\mathbf{z}))$$

A Lagrangian Perspective of Latent Variable Generative Models, Zhao, Song, Ermon. UAI 2018.

$$\min I_q(\mathbf{u}, \mathbf{z})$$

$$I_q(\mathbf{z}; \mathbf{u}) \leq I_q(\mathbf{z}; \mathbf{x}, \mathbf{u}) = \mathbb{E}_{q(\mathbf{x},\mathbf{u})} D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{u}) \| p(\mathbf{z})) - D_{\mathrm{KL}}(q_\phi(\mathbf{z}) \| p(\mathbf{z}))$$

Add additional variable

A Lagrangian Perspective of Latent Variable Generative Models, Zhao, Song, Ermon. UAI 2018.

$$\min I_q(\mathbf{u}, \mathbf{z})$$

$$I_q(\mathbf{z}; \mathbf{u}) \leq I_q(\mathbf{z}; \mathbf{x}, \mathbf{u}) = \mathbb{E}_{q(\mathbf{x},\mathbf{u})} D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{u}) \| p(\mathbf{z})) - D_{\mathrm{KL}}(q_\phi(\mathbf{z}) \| p(\mathbf{z}))$$

Add additional variable

KL divergence > 0

A Lagrangian Perspective of Latent Variable Generative Models, Zhao, Song, Ermon. UAI 2018.

$$\min I_q(\mathbf{u}, \mathbf{z})$$

$$I_q(\mathbf{z}; \mathbf{u}) \le I_q(\mathbf{z}; \mathbf{x}, \mathbf{u}) = \mathbb{E}_{q(\mathbf{x},\mathbf{u})} D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{u}) \| p(\mathbf{z})) - D_{\mathrm{KL}}(q_\phi(\mathbf{z}) \| p(\mathbf{z}))$$

Add additional variable

"Rate"
Upper Bound

KL divergence > 0

A Lagrangian Perspective of Latent Variable Generative Models, Zhao, Song, Ermon. UAI 2018.

$$\min I_q(\mathbf{u}, \mathbf{z})$$

$$I_q(\mathbf{z};\mathbf{u}) \leq I_q(\mathbf{z};\mathbf{x},\mathbf{u}) = \mathbb{E}_{q(\mathbf{x},\mathbf{u})} D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x},\mathbf{u}) \| p(\mathbf{z})) - D_{\mathrm{KL}}(q_\phi(\mathbf{z}) \| p(\mathbf{z}))$$

Add additional variable

"Rate"
Upper Bound

KL divergence > 0

$I_q(\mathbf{x};\mathbf{z}|\mathbf{u})$ ————————

A Lagrangian Perspective of Latent Variable Generative Models, Zhao, Song, Ermon. UAI 2018.

$$\min I_q(\mathbf{u}, \mathbf{z})$$

$$I_q(\mathbf{z}; \mathbf{u}) \leq I_q(\mathbf{z}; \mathbf{x}, \mathbf{u}) = \mathbb{E}_{q(\mathbf{x},\mathbf{u})} D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{u}) \| p(\mathbf{z})) - D_{\mathrm{KL}}(q_\phi(\mathbf{z}) \| p(\mathbf{z}))$$

Add additional variable

"Rate"
Upper Bound

KL divergence > 0

A Lagrangian Perspective of Latent Variable Generative Models, Zhao, Song, Ermon. UAI 2018.

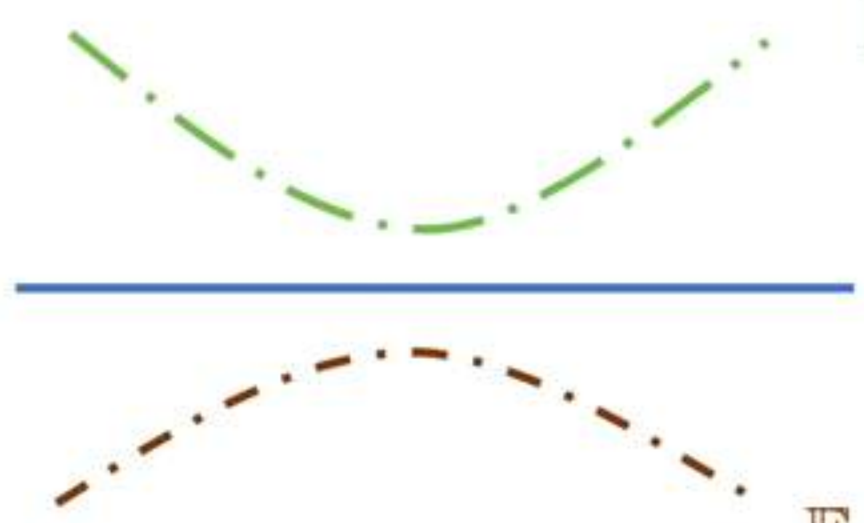
$$\min I_q(\mathbf{u}, \mathbf{z})$$

$$I_q(\mathbf{z}; \mathbf{u}) \leq I_q(\mathbf{z}; \mathbf{x}, \mathbf{u}) = \mathbb{E}_{q(\mathbf{x},\mathbf{u})} D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{u}) \| p(\mathbf{z})) - D_{\mathrm{KL}}(q_\phi(\mathbf{z}) \| p(\mathbf{z}))$$

Add additional variable

"Rate"
Upper Bound

KL divergence > 0



A Lagrangian Perspective of Latent Variable
Generative Models, Zhao, Song, Ermon. UAI 2018.

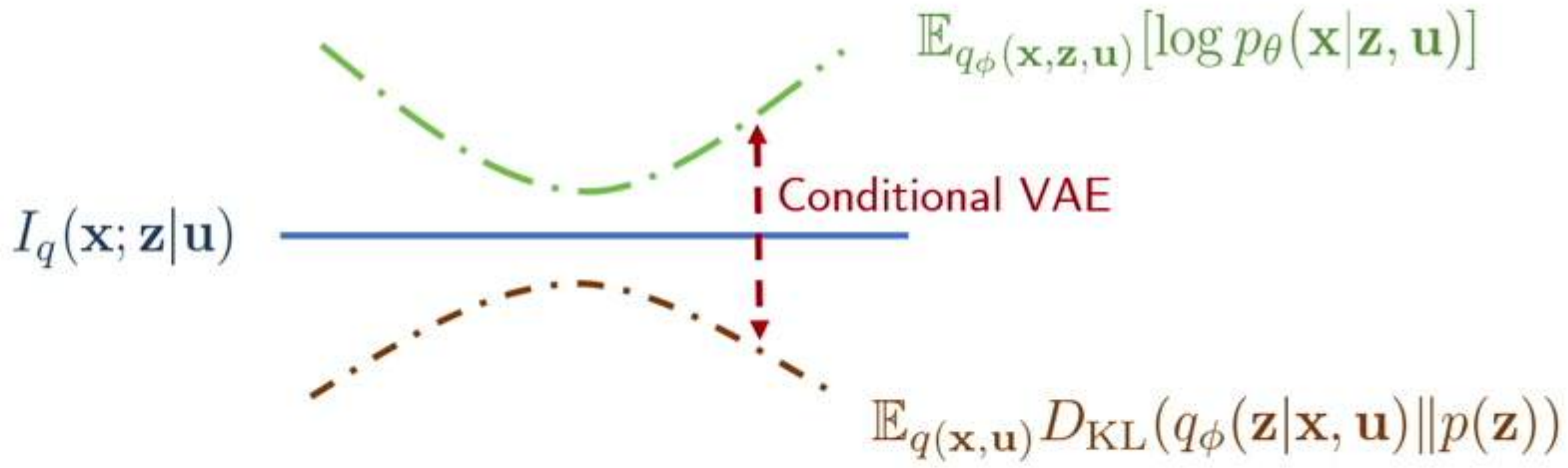
$$\min I_q(\mathbf{u}, \mathbf{z})$$

$$I_q(\mathbf{z}; \mathbf{u}) \leq I_q(\mathbf{z}; \mathbf{x}, \mathbf{u}) = \mathbb{E}_{q(\mathbf{x},\mathbf{u})} D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x}, \mathbf{u}) \| p(\mathbf{z})) - D_{\mathrm{KL}}(q_\phi(\mathbf{z}) \| p(\mathbf{z}))$$

Add additional variable

"Rate"
Upper Bound

KL divergence > 0



A Lagrangian Perspective of Latent Variable Generative Models, Zhao, Song, Ermon. UAI 2018.

$$\min I_q(\mathbf{u}, \mathbf{z})$$

$$\min I_q(\mathbf{u}, \mathbf{z})$$

The “rate” upper bound is not tight!

Tighter version with some $p(\mathbf{u})$:

$$\min I_q(\mathbf{u}, \mathbf{z})$$

The "rate" upper bound is not tight!

Tighter version with some $p(\mathbf{u})$:

$$I_q(\mathbf{z}; \mathbf{u}) = \mathbb{E}_{q_\phi(\mathbf{z})} D_{\mathrm{KL}}(q_\phi(\mathbf{u}|\mathbf{z}) \| p(\mathbf{u})) - D_{\mathrm{KL}}(q(\mathbf{u}) \| p(\mathbf{u}))$$

$$\min I_q(\mathbf{u}, \mathbf{z})$$

The "rate" upper bound is not tight!

Tighter version with some $p(\mathbf{u})$:

$$I_q(\mathbf{z}; \mathbf{u}) = \mathbb{E}_{q_\phi(\mathbf{z})} D_{\mathrm{KL}}(q_\phi(\mathbf{u}|\mathbf{z}) \| p(\mathbf{u})) - D_{\mathrm{KL}}(q(\mathbf{u}) \| p(\mathbf{u}))$$

Upper bound, but we don't know $q_\phi(\mathbf{u}|\mathbf{z})$

$$\min I_q(\mathbf{u}, \mathbf{z})$$

The "rate" upper bound is not tight!

Tighter version with some $p(\mathbf{u})$:

$$I_q(\mathbf{z}; \mathbf{u}) = \mathbb{E}_{q_\phi(\mathbf{z})} D_{\mathrm{KL}}(q_\phi(\mathbf{u}|\mathbf{z}) \| p(\mathbf{u})) - D_{\mathrm{KL}}(q(\mathbf{u}) \| p(\mathbf{u}))$$

Upper bound, but we don't know $q_\phi(\mathbf{u}|\mathbf{z})$

Approximate $q_\phi(\mathbf{u}|\mathbf{z})$ with $p_\psi(\mathbf{u}|\mathbf{z})$ by maximum likelihood!

“Adversarial Training”

“Adversarial Training”

Approximate $q_\phi(\mathbf{u}|\mathbf{z})$ with $p_\psi(\mathbf{u}|\mathbf{z})$ by maximum likelihood!

“Adversarial Training”

Approximate $q_\phi(\mathbf{u}|\mathbf{z})$ with $p_\psi(\mathbf{u}|\mathbf{z})$ by maximum likelihood!

$$\min_{\psi} \mathbb{E}_{q_\phi(\mathbf{z})} D_{\mathrm{KL}}(q_\phi(\mathbf{u}|\mathbf{z}) \| p_\psi(\mathbf{u}|\mathbf{z}))$$

"Adversarial Training"

Approximate $q_\phi(\mathbf{u}|\mathbf{z})$ with $p_\psi(\mathbf{u}|\mathbf{z})$ by maximum likelihood!

$$\min_\psi \mathbb{E}_{q_\phi(\mathbf{z})} D_{\mathrm{KL}}(q_\phi(\mathbf{u}|\mathbf{z}) \| p_\psi(\mathbf{u}|\mathbf{z}))$$

Replace q with p, we have

"Adversarial Training"

Approximate $q_\phi(\mathbf{u}|\mathbf{z})$ with $p_\psi(\mathbf{u}|\mathbf{z})$ by maximum likelihood!

$$\min_\psi \mathbb{E}_{q_\phi(\mathbf{z})} D_{\mathrm{KL}}(q_\phi(\mathbf{u}|\mathbf{z}) \| p_\psi(\mathbf{u}|\mathbf{z}))$$

Replace q with p, we have

$$\begin{aligned}
&\mathbb{E}_{q_\phi(\mathbf{z},\mathbf{u})}[\log p_\psi(\mathbf{u}|\mathbf{z}) - \log p(\mathbf{u})] \\
&= \mathbb{E}_{q_\phi(\mathbf{z})}[D_{\mathrm{KL}}(q_\phi(\mathbf{u}|\mathbf{z}) \| p(\mathbf{u})) - D_{\mathrm{KL}}(q_\phi(\mathbf{u}|\mathbf{z}) \| p_\psi(\mathbf{u}|\mathbf{z}))] \\
&\leq \mathbb{E}_{q_\phi(\mathbf{z})} D_{\mathrm{KL}}(q_\phi(\mathbf{u}|\mathbf{z}) \| p(\mathbf{u}))
\end{aligned}$$

"Adversarial Training"

Approximate $q_\phi(\mathbf{u}|\mathbf{z})$ with $p_\psi(\mathbf{u}|\mathbf{z})$ by maximum likelihood!

$$\min_\psi \mathbb{E}_{q_\phi(\mathbf{z})} D_{\mathrm{KL}}(q_\phi(\mathbf{u}|\mathbf{z}) \| p_\psi(\mathbf{u}|\mathbf{z}))$$

Replace q with p, we have

$$\mathbb{E}_{q_\phi(\mathbf{z},\mathbf{u})}[\log p_\psi(\mathbf{u}|\mathbf{z}) - \log p(\mathbf{u})]$$
$$= \mathbb{E}_{q_\phi(\mathbf{z})}[D_{\mathrm{KL}}(q_\phi(\mathbf{u}|\mathbf{z}) \| p(\mathbf{u})) - D_{\mathrm{KL}}(q_\phi(\mathbf{u}|\mathbf{z}) \| p_\psi(\mathbf{u}|\mathbf{z}))]$$
$$\leq \mathbb{E}_{q_\phi(\mathbf{z})} D_{\mathrm{KL}}(q_\phi(\mathbf{u}|\mathbf{z}) \| p(\mathbf{u}))$$

"Maximum likelihood"

"Adversarial Training"

Approximate $q_\phi(\mathbf{u}|\mathbf{z})$ with $p_\psi(\mathbf{u}|\mathbf{z})$ by maximum likelihood!

$$\min_\psi \mathbb{E}_{q_\phi(\mathbf{z})} D_{\mathrm{KL}}(q_\phi(\mathbf{u}|\mathbf{z}) \| p_\psi(\mathbf{u}|\mathbf{z}))$$

Replace q with p, we have

$$\mathbb{E}_{q_\phi(\mathbf{z},\mathbf{u})}[\log p_\psi(\mathbf{u}|\mathbf{z}) - \log p(\mathbf{u})]$$

"Lower bound to upper bound"

$$= \mathbb{E}_{q_\phi(\mathbf{z})}[D_{\mathrm{KL}}(q_\phi(\mathbf{u}|\mathbf{z}) \| p(\mathbf{u})) - D_{\mathrm{KL}}(q_\phi(\mathbf{u}|\mathbf{z}) \| p_\psi(\mathbf{u}|\mathbf{z}))]$$

$$\leq \mathbb{E}_{q_\phi(\mathbf{z})} D_{\mathrm{KL}}(q_\phi(\mathbf{u}|\mathbf{z}) \| p(\mathbf{u}))$$

"Maximum likelihood"

"Adversarial Training"

Approximate $q_\phi(\mathbf{u}|\mathbf{z})$ with $p_\psi(\mathbf{u}|\mathbf{z})$ by maximum likelihood!

$$\min_{\psi} \mathbb{E}_{q_\phi(\mathbf{z})} D_{\mathrm{KL}}(q_\phi(\mathbf{u}|\mathbf{z}) \| p_\psi(\mathbf{u}|\mathbf{z}))$$

Replace q with p, we have

$$\mathbb{E}_{q_\phi(\mathbf{z},\mathbf{u})}[\log p_\psi(\mathbf{u}|\mathbf{z}) - \log p(\mathbf{u})]$$ "Lower bound to upper bound"

$$= \mathbb{E}_{q_\phi(\mathbf{z})}[D_{\mathrm{KL}}(q_\phi(\mathbf{u}|\mathbf{z}) \| p(\mathbf{u})) - D_{\mathrm{KL}}(q_\phi(\mathbf{u}|\mathbf{z}) \| p_\psi(\mathbf{u}|\mathbf{z}))]$$

$$\leq \mathbb{E}_{q_\phi(\mathbf{z})} D_{\mathrm{KL}}(q_\phi(\mathbf{u}|\mathbf{z}) \| p(\mathbf{u}))$$ "Maximum likelihood"

Maximum likelihood --> Gap with upper bound = 0!

“Adversarial Training” Intuition

“Adversarial Training” Intuition

Classifier: predict u from z.

“Adversarial Training” Intuition

Classifier: predict u from z.

Representation: prevent classifier to predict u.

"Adversarial Training" Intuition

Classifier: predict u from z.

Representation: prevent classifier to predict u.

- Adversarial training but not a GAN!

“Adversarial Training” Intuition

Classifier: predict u from z.

Representation: prevent classifier to predict u.

- Adversarial training but not a GAN!
- Allows for any type of u (as opposed to binary)

“Tractable” Objective

"Tractable" Objective

$$\begin{aligned}\min_{\theta,\phi}\max_{\psi\in\Psi} \quad & \mathcal{L}_r = -\mathbb{E}_{q_\phi(\mathbf{x},\mathbf{z},\mathbf{u})}[\log p_\theta(\mathbf{x}|\mathbf{z},\mathbf{u})] \\ \text{s.t.} \quad & C_1 = \mathbb{E}_{q(\mathbf{x},\mathbf{u})} D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x},\mathbf{u}) \| p(\mathbf{z})) < \epsilon_1 \\ & C_2 = \mathbb{E}_{q_\phi(\mathbf{z},\mathbf{u})}[\log p_\psi(\mathbf{u}|\mathbf{z}) - \log p(\mathbf{u})] < \epsilon_2\end{aligned}$$

Dual Formulation

Dual Formulation

$$\arg\min_{\theta,\phi}\max_{\psi}\mathcal{L}_r + \lambda_1(C_1 - \epsilon_1) + \lambda_2(C_2 - \epsilon_2)$$

Dual Formulation

$$\arg\min_{\theta,\phi}\max_{\psi}\mathcal{L}_r + \lambda_1(C_1 - \epsilon_1) + \lambda_2(C_2 - \epsilon_2)$$

Existing works optimize this objective with fixed lambda!

Dual Formulation

$$\arg\min_{\theta,\phi}\max_{\psi} \mathcal{L}_r + \lambda_1(C_1 - \epsilon_1) + \lambda_2(C_2 - \epsilon_2)$$

Existing works optimize this objective with fixed lambda!

- Variational Fair Autoencoder: $\lambda_1 = 1$; MMD on two groups of z.

Dual Formulation

$$\arg\min_{\theta,\phi}\max_{\psi}\mathcal{L}_r + \lambda_1(C_1 - \epsilon_1) + \lambda_2(C_2 - \epsilon_2)$$

Existing works optimize this objective with fixed lambda!

- Variational Fair Autoencoder: $\lambda_1 = 1$; MMD on two groups of z.
- Adversarial Censoring: $\lambda_1 = 0$; GAN on two groups of x.

Dual Formulation

$$\arg\min_{\theta,\phi}\max_{\psi} \mathcal{L}_r + \lambda_1(C_1 - \epsilon_1) + \lambda_2(C_2 - \epsilon_2)$$

Existing works optimize this objective with fixed lambda!

- Variational Fair Autoencoder: $\lambda_1 = 1$; MMD on two groups of z.
- Adversarial Censoring: $\lambda_1 = 0$; GAN on two groups of x.
- Fair and Transferrable Representations: $\lambda_1 = 0$; GAN on two groups of z.

Dual Optimization

Dual Optimization

Straightforward to use dual optimization

Dual Optimization

Straightforward to use dual optimization

$$\max_{\lambda_1,\lambda_2\geq 0}\min_{\theta,\phi}\max_{\psi}\mathcal{L}=\mathcal{L}_r+\lambda_1^\top(C_1-\epsilon_1)+\lambda_2^\top(C_2-\epsilon_2)$$

Dual Optimization

Straightforward to use dual optimization

$$\max_{\lambda_1,\lambda_2\geq 0}\min_{\theta,\phi}\max_{\psi}\mathcal{L}=\mathcal{L}_r+\lambda_1^\top(C_1-\epsilon_1)+\lambda_2^\top(C_2-\epsilon_2)$$

If constraint is violated, increase its weight

Dual Optimization

Straightforward to use dual optimization

$$\max_{\lambda_1,\lambda_2\geq 0}\min_{\theta,\phi}\max_{\psi}\mathcal{L}=\mathcal{L}_r+\lambda_1^\top(C_1-\epsilon_1)+\lambda_2^\top(C_2-\epsilon_2)$$

If constraint is violated, increase its weight

- Find the trade-off between fairness and expressiveness!

Dual Optimization

Straightforward to use dual optimization

$$\max_{\lambda_1, \lambda_2 \geq 0} \min_{\theta, \phi} \max_{\psi} \mathcal{L} = \mathcal{L}_r + \lambda_1^\top (C_1 - \epsilon_1) + \lambda_2^\top (C_2 - \epsilon_2)$$

If constraint is violated, increase its weight

- Find the trade-off between fairness and expressiveness!
- Particularly useful with multiple fairness notions!

Dual Optimization

Straightforward to use dual optimization

$$\max_{\lambda_1,\lambda_2\geq 0}\min_{\theta,\phi}\max_{\psi}\mathcal{L}=\mathcal{L}_r+\lambda_1^\top(C_1-\epsilon_1)+\lambda_2^\top(C_2-\epsilon_2)$$

If constraint is violated, increase its weight

- Find the trade-off between fairness and expressiveness!
- Particularly useful with multiple fairness notions!
- "Find solution that is reasonable under multiple notions"

Dual Optimization

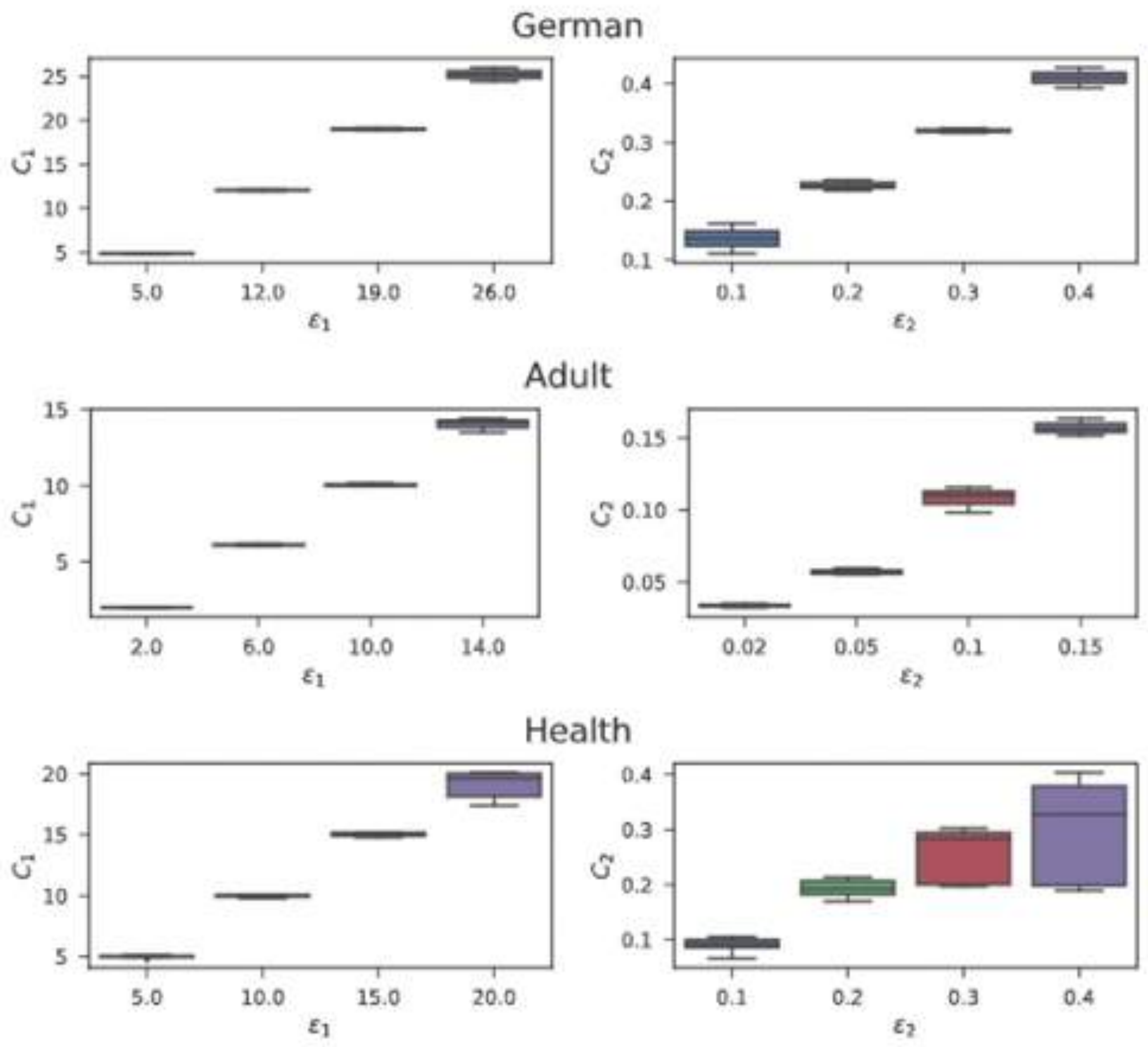
Straightforward to use dual optimization

$$\max_{\lambda_1,\lambda_2 \geq 0} \min_{\theta,\phi} \max_{\psi} \mathcal{L} = \mathcal{L}_r + \lambda_1^\top (C_1 - \epsilon_1) + \lambda_2^\top (C_2 - \epsilon_2)$$

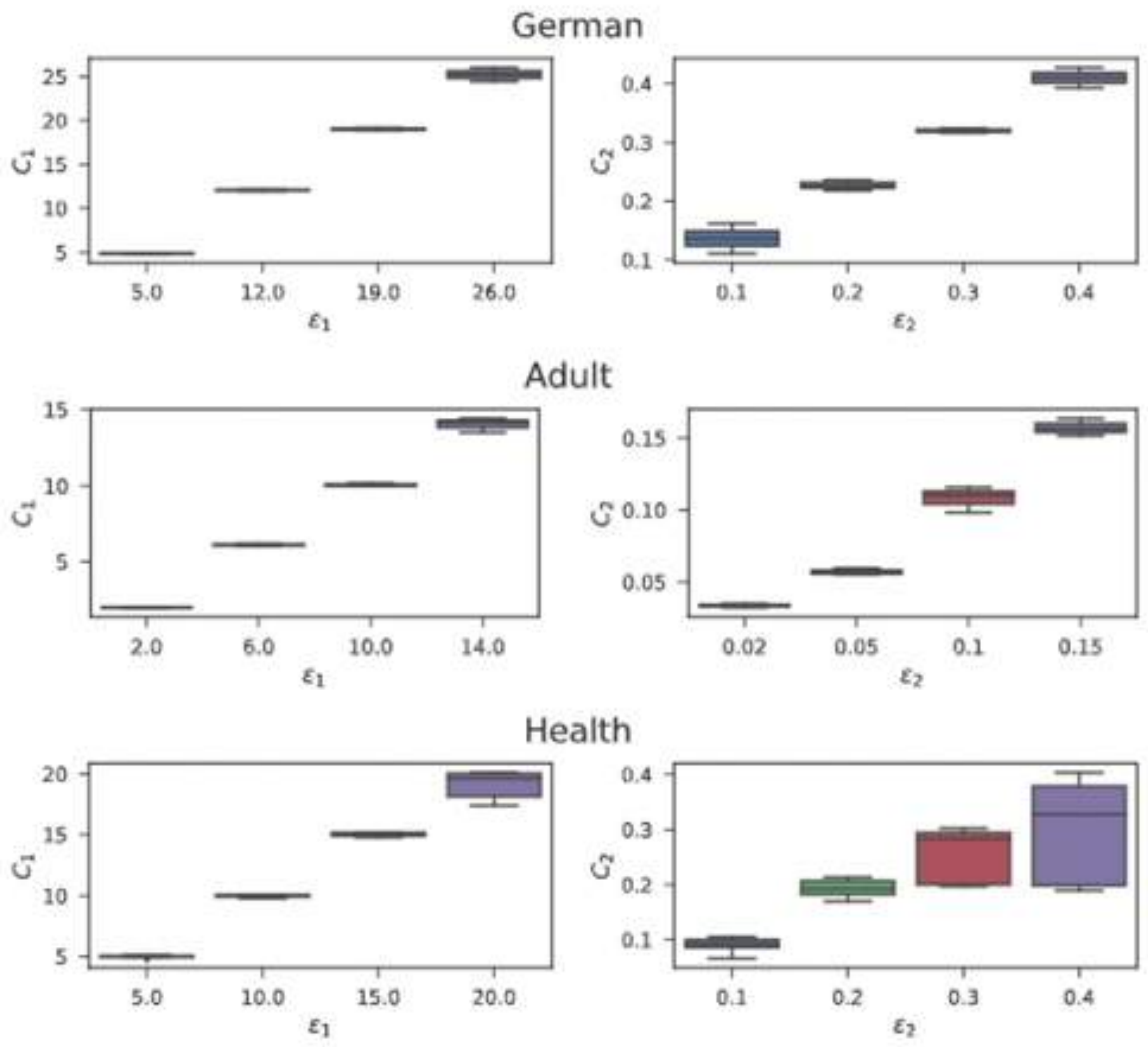
If constraint is violated, increase its weight

- Find the trade-off between fairness and expressiveness!
- Particularly useful with multiple fairness notions!
- "Find solution that is reasonable under multiple notions"
- Allows user to control level of "unfairness" directly

Experiments



Experiments



Dual optimization can find feasible solutions!

Learning Better Representations

Learning Better Representations

- Search for most expressive representations under certain fairness constraints.

Learning Better Representations

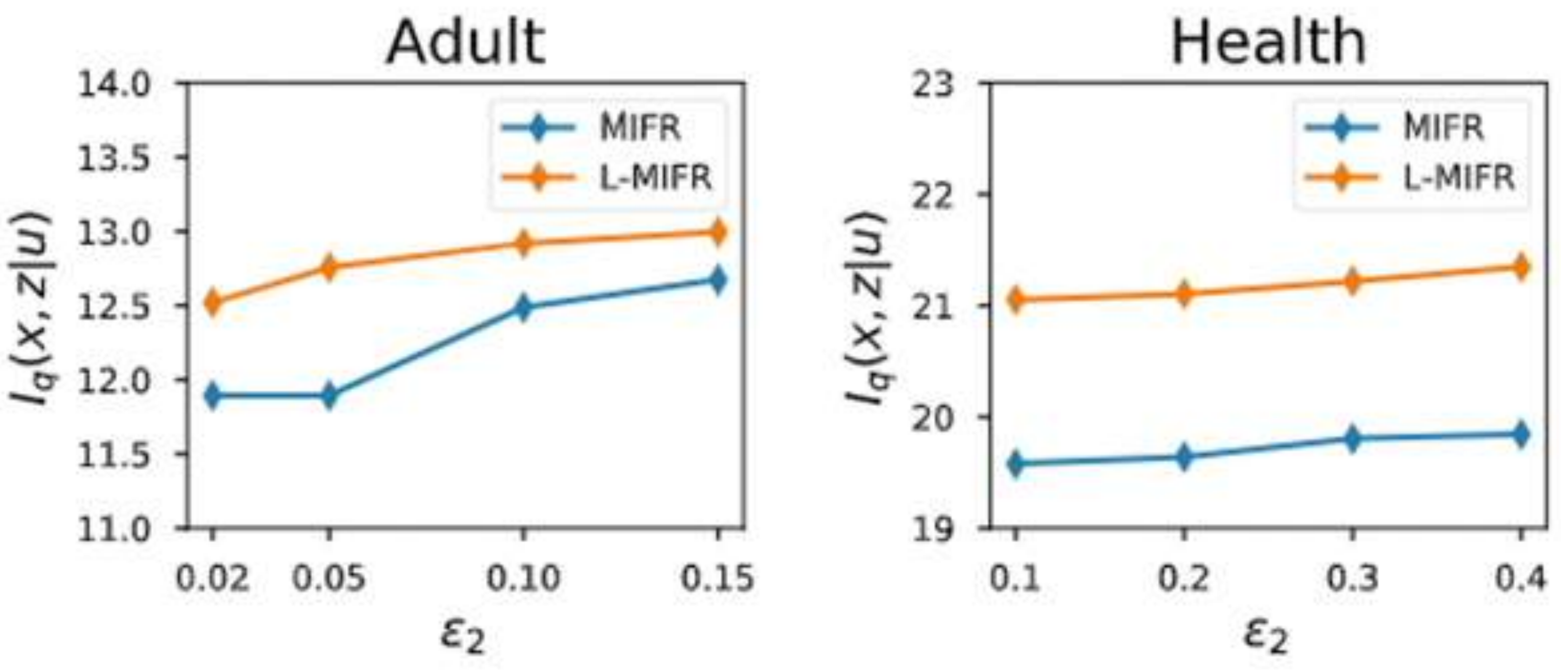
- Search for most expressive representations under certain fairness constraints.
- MIFR: fixed multipliers, grid search over 5x5 hyperparameters, find most "expressive" feasible solution.

Learning Better Representations

- Search for most expressive representations under certain fairness constraints.
- MIFR: fixed multipliers, grid search over 5x5 hyperparameters, find most "expressive" feasible solution.
- L-MIFR: train Lagrange multipliers on-the-fly, run once.

Learning Better Representations

- Search for most expressive representations under certain fairness constraints.
- MIFR: fixed multipliers, grid search over 5x5 hyperparameters, find most "expressive" feasible solution.
- L-MIFR: train Lagrange multipliers on-the-fly, run once.



Learning Better Representations

Learning Better Representations

- Search for most expressive representations under multiple fairness constraints.
  - Demographic parity, Equalized odds, Equalized opportunity

Learning Better Representations

- Search for most expressive representations under multiple fairness constraints.
  - Demographic parity, Equalized odds, Equalized opportunity
- MIFR: greedy hyperparameter search, 12 runs.

Learning Better Representations

- Search for most expressive representations under multiple fairness constraints.
  - Demographic parity, Equalized odds, Equalized opportunity
- MIFR: greedy hyperparameter search, 12 runs.
- L-MIFR: train Lagrange multipliers on-the-fly, run once.

Learning Better Representations

- Search for most expressive representations under multiple fairness constraints.
  - Demographic parity, Equalized odds, Equalized opportunity
- MIFR: greedy hyperparameter search, 12 runs.
- L-MIFR: train Lagrange multipliers on-the-fly, run once.

	$I_q(\mathbf{x};\mathbf{z}\mid\mathbf{u})$	C_1	I_{DP}	I_{EO}	I_{EOpp}
MIFR	9.34	**9.39**	0.09	0.10	0.07
L-MIFR	**9.94**	9.95	**0.08**	**0.09**	**0.04**

Learning Better Representations

- Search for most expressive representations under multiple fairness constraints.
  - Demographic parity, Equalized odds, Equalized opportunity
- MIFR: greedy hyperparameter search, 12 runs.
- L-MIFR: train Lagrange multipliers on-the-fly, run once.

	$I_q(\mathbf{x};\mathbf{z}\|\mathbf{u})$	C_1	I_{DP}	I_{EO}	I_{EOpp}
MIFR	9.34	**9.39**	0.09	0.10	0.07
L-MIFR	**9.94**	9.95	**0.08**	**0.09**	**0.04**

L-MIFR learns better representations!

Summary

Summary

- Information-theoretic view of fair representation learning

Summary

- Information-theoretic view of fair representation learning
- Connection with VAEs and GANs

Summary

- Information-theoretic view of fair representation learning
- Connection with VAEs and GANs
- Dual optimization avoids tedious hyperparameter tuning and learns trade-offs

Summary

- Information-theoretic view of fair representation learning
- Connection with VAEs and GANs
- Dual optimization avoids tedious hyperparameter tuning and learns trade-offs
- Better representation with less compute