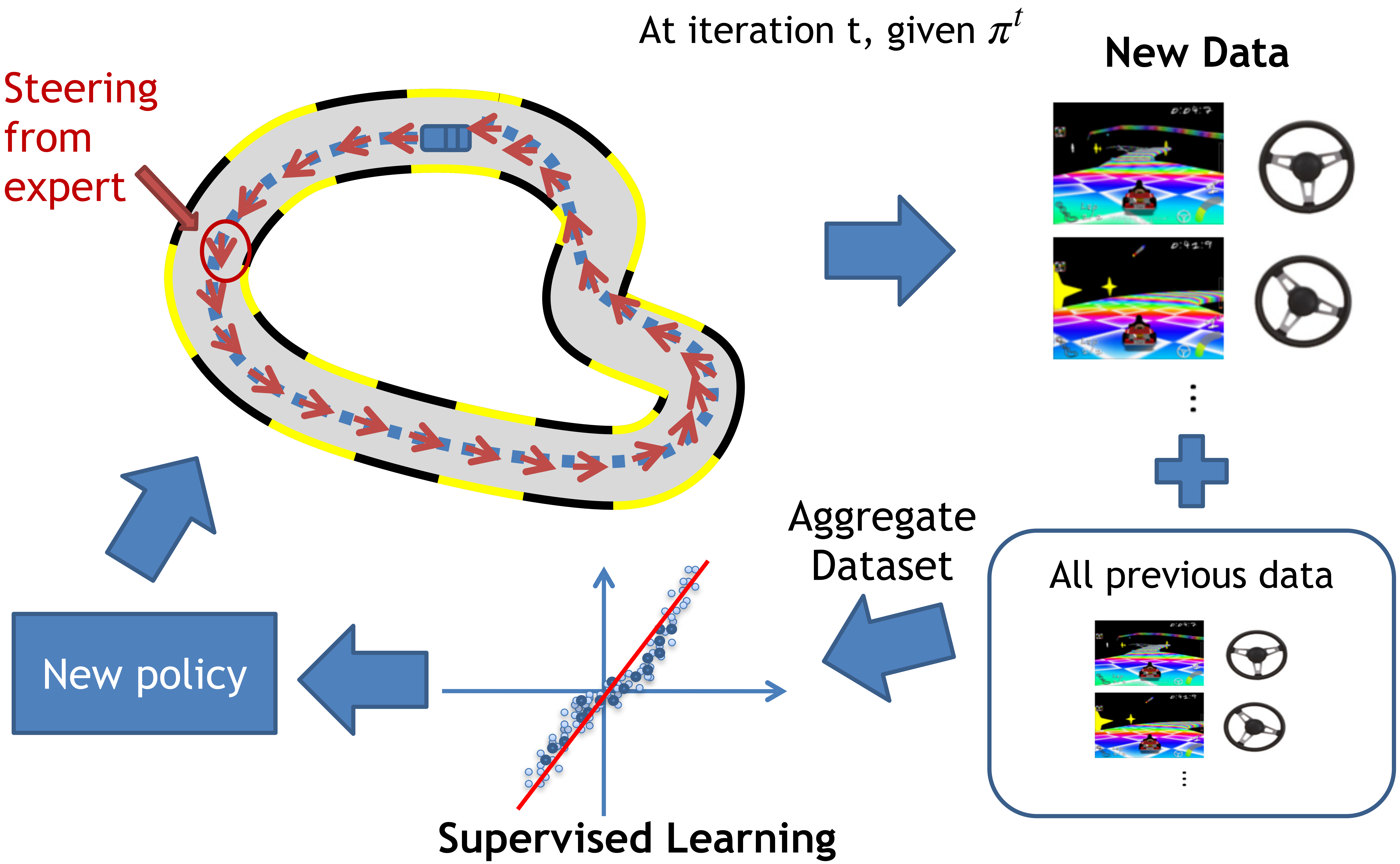
Maximum Entropy IRL

DAgger Recap



Data Aggregation = Follow-the-Regularized-Leader Online Learner

DAgger Performance Recap:

DAgger finds a policy $\widehat{\pi}$ such that it **matches to $\pi^\star$ under its own $d_\mu^{\widehat{\pi}}$**

$$\mathbb{E}_{s \sim d_\mu^{\widehat{\pi}}} \left[\mathbf{1}\{ \widehat{\pi}(s) \neq \pi^\star(s) \} \right] \leq \epsilon_{reg} = O(1/\sqrt{T})$$

DAgger Performance Recap:

DAgger finds a policy $\widehat{\pi}$ such that it **matches to $\pi^\star$ under its own $d_\mu^{\widehat{\pi}}$**

$$\mathbb{E}_{s \sim d_\mu^{\widehat{\pi}}}\left[\mathbf{1}\{\widehat{\pi}(s) \neq \pi^\star(s)\}\right] \leq \epsilon_{reg} = O(1/\sqrt{T})$$

If expert herself can **quickly recover from a deviation**, i.e., $|Q^{\pi^\star}(s,a) - V^{\pi^\star}(s)|$ is small for all s,

$$V^{\pi^\star} - V^{\pi^t} \leq O\left(\frac{1}{1-\gamma} \cdot \epsilon_{reg}\right)$$

DAgger Performance Recap:

DAgger finds a policy $\widehat{\pi}$ such that it **matches to $\pi^\star$ under its own $d_\mu^{\widehat{\pi}}$**

$$\mathbb{E}_{s\sim d_\mu^{\widehat{\pi}}}\left[\mathbf{1}\{\widehat{\pi}(s) \neq \pi^\star(s)\}\right] \leq \epsilon_{reg} = O(1/\sqrt{T})$$

If expert herself can **quickly recover from a deviation**, i.e., $|Q^{\pi^\star}(s,a) - V^{\pi^\star}(s)|$ is small for all s,

$$V^{\pi^\star} - V^{\pi^t} \leq O\left(\frac{1}{1-\gamma}\cdot\epsilon_{reg}\right)$$

This is a significant improvement over BC in both theory and practice

Plan for Today:

1. The principle of Maximum Entropy

2. Constrained Optimization

2. The Algorithm: Maximum Entropy Inverse RL

Setting

Finite horizon MDP $\mathscr{M} = \{S, A, H, c, P, \mu, \pi^\star\}$

Setting

Finite horizon MDP $\mathscr{M} = \{S, A, H, c, P, \mu, \pi^\star\}$

(1) Ground truth cost $c(s, a)$ is unknown;
(2) assume expert is the optimal policy $\pi^\star$ of the cost c
(3) **transition P is known**

Setting

Finite horizon MDP $\mathscr{M} = \{S, A, H, c, P, \mu, \pi^\star\}$

(1) Ground truth cost $c(s, a)$ is unknown;

(2) assume expert is the optimal policy $\pi^\star$ of the cost c

(3) **transition P is known**

We have a dataset $\mathscr{D} = (s_i^\star, a_i^\star)_{i=1}^M \sim d^{\pi^\star}$

Setting

Finite horizon MDP $\mathscr{M} = \{S, A, H, c, P, \mu, \pi^\star\}$

(1) Ground truth cost $c(s, a)$ is unknown;
(2) assume expert is the optimal policy $\pi^\star$ of the cost c
(3) **transition P is known**

We have a dataset $\mathscr{D} = (s_i^\star, a_i^\star)_{i=1}^M \sim d^{\pi^\star}$

Key Assumption on cost:
$c(s, a) = \langle \theta^\star, \phi(s, a) \rangle$, **linear w.r.t feature** $\phi(s, a)$

Running Example: Define feature map

Key Assumption on cost:

$c(s,a) = \langle \theta^\star, \phi(s,a) \rangle$, **linear wrt feature** $\phi(s,a)$

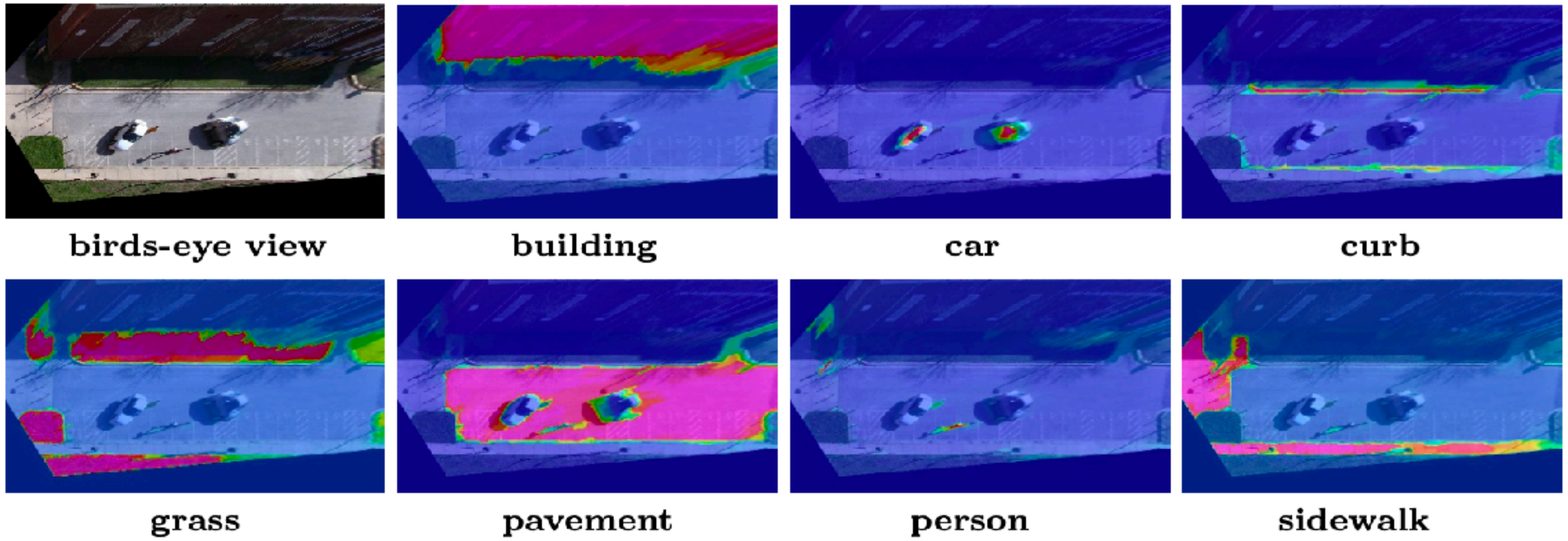


Fig. 4. Classifier feature response maps. Top left is the original image.

Running Example: Define feature map

Key Assumption on cost:

$c(s,a) = \langle \theta^\star, \phi(s,a) \rangle$, **linear wrt feature** $\phi(s,a)$

State s: pixel or a group of neighboring pixels in image)

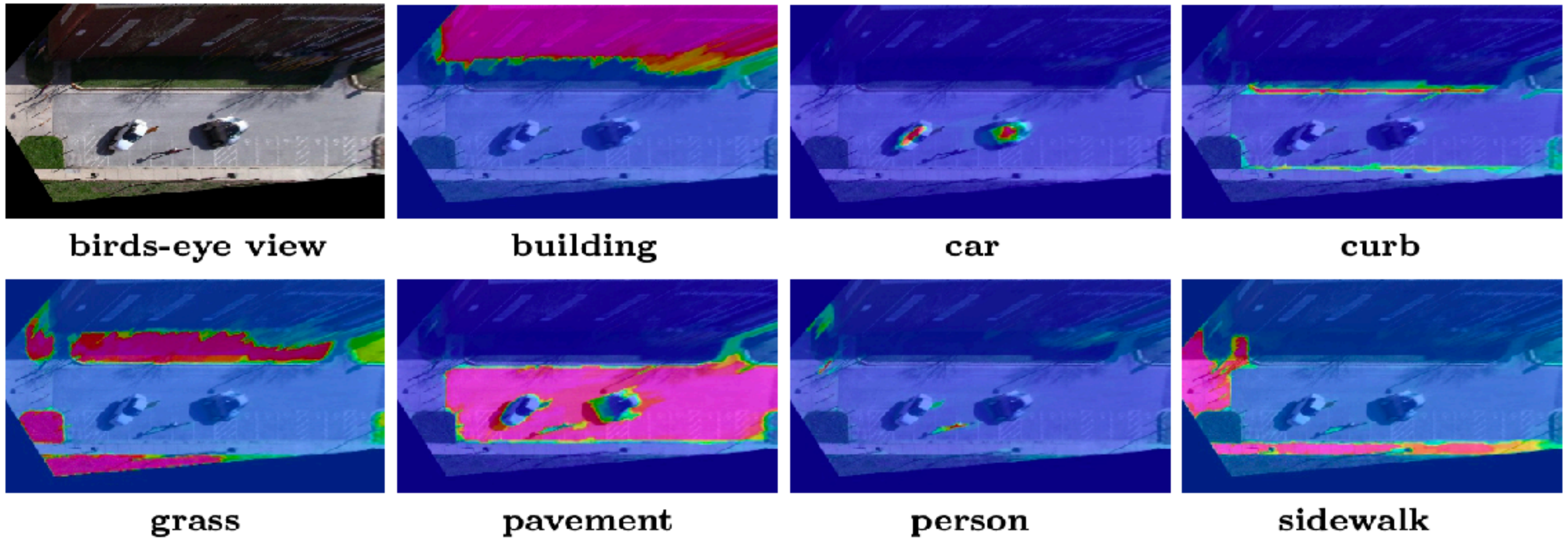


Fig. 4. Classifier feature response maps. Top left is the original image.

Running Example: Define feature map

Key Assumption on cost:

$c(s, a) = \langle \theta^\star, \phi(s, a) \rangle$, **linear wrt feature** $\phi(s, a)$

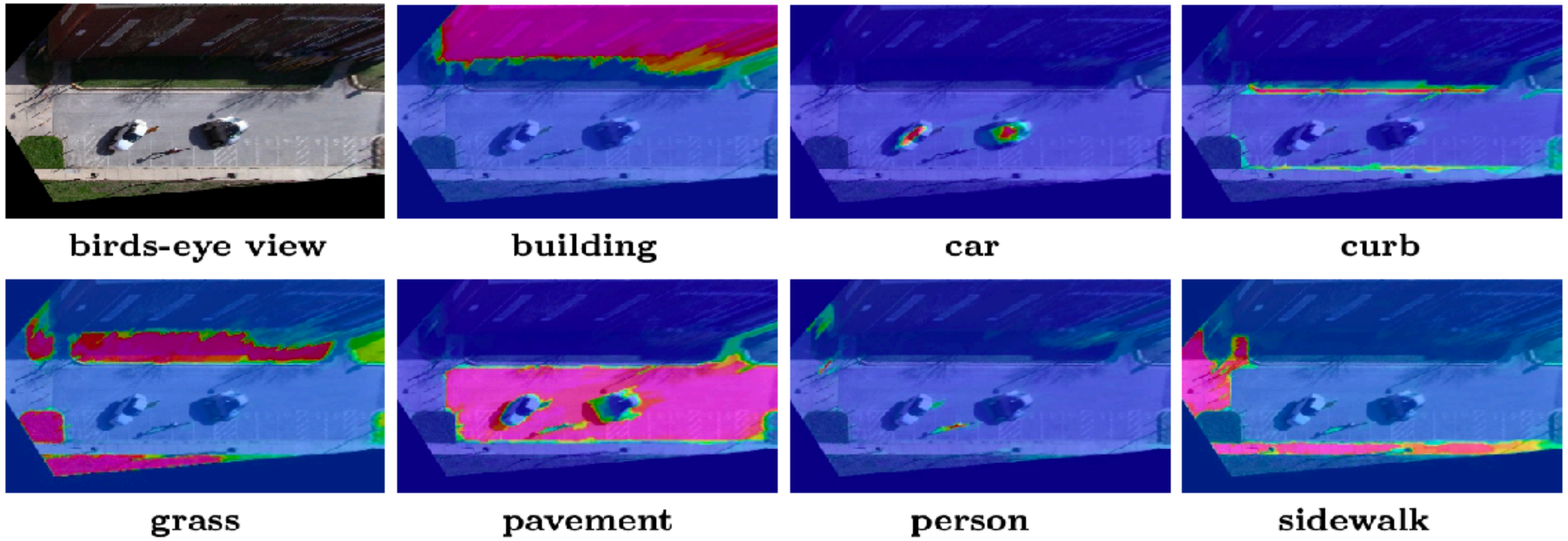


Fig. 4. Classifier feature response maps. Top left is the original image.

State s: pixel or a group of neighboring pixels in image)

$$\phi(s,a) = \begin{bmatrix} \mathbb{P}(\text{pixels being building}) \\ \mathbb{P}(\text{pixels being grass}) \\ \mathbb{P}(\text{pixels being sidewalk}) \\ \mathbb{P}(\text{pixels being car}) \\ \cdots \end{bmatrix}$$

Running Example: Define feature map

Key Assumption on cost:

$c(s,a) = \langle \theta^\star, \phi(s,a) \rangle$, **linear wrt feature** $\phi(s,a)$

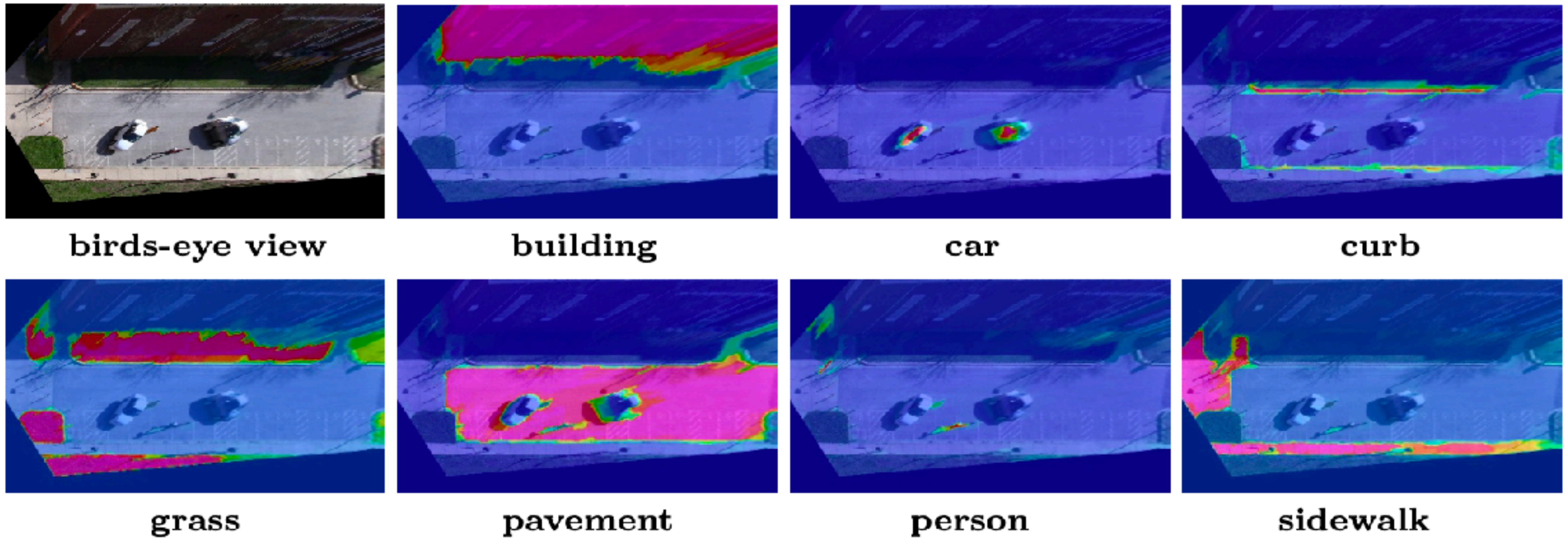


Fig. 4. Classifier feature response maps. Top left is the original image.

State s: pixel or a group of neighboring pixels in image)

$$\phi(s,a) = \begin{bmatrix} \mathbb{P}(\text{pixels being building}) \\ \mathbb{P}(\text{pixels being grass}) \\ \mathbb{P}(\text{pixels being sidewalk}) \\ \mathbb{P}(\text{pixels being car}) \\ \cdots \end{bmatrix}$$

Maybe colliding with cars or buildings has **high** cost, but walking on sideway or grass has **low** cost

Notation on Distributions

$\mathbb{P}_h^\pi(s, a; \mu)$: probability of visiting (s, a) at time step h following π

$d_\mu^\pi(s, a) = \sum_{h=0}^{H-1} \mathbb{P}_h^\pi(s, a; \mu)/H$: average state-action distribution

$\rho^\pi(\tau) := \mu_0(s_0)\pi(a_0 \mid s_0)P(s_1 \mid s_0, a_0)\pi(a_1 \mid s_1)\ldots\pi(a_{H-1} \mid s_{H-1})P(s_H \mid s_{H-1}, a_{H-1})$:
Likelihood of the trajectory τ under π, i.e., the prob of π generating τ

Detour: Principle of Maximum Entropy

Definition of the Entropy of a distribution:

Detour: Principle of Maximum Entropy

Definition of the Entropy of a distribution:

Given a distribution $P \in \Delta(X)$, the entropy is defined as:

$$\text{Entropy}(P) = -\sum_x P(x) \cdot \ln P(x)$$

Detour: Principle of Maximum Entropy

Definition of the Entropy of a distribution:

Given a distribution $P \in \Delta(X)$, the entropy is defined as:

$$\text{Entropy}(P) = -\sum_x P(x) \cdot \ln P(x)$$

Higher entropy means more uncertainty, for instance:

Detour: Principle of Maximum Entropy

Definition of the Entropy of a distribution:

Given a distribution $P \in \Delta(X)$, the entropy is defined as:

$$\text{Entropy}(P) = -\sum_{x} P(x) \cdot \ln P(x)$$

Higher entropy means more uncertainty, for instance:

Uniform distribution has the highest entropy,

i.e., $\text{Entropy}(U(X)) = -\sum_{x} (1/|X|)\ln(1/|X|) = \ln(|X|)$

Detour: Principle of Maximum Entropy

Definition of the Entropy of a distribution:

Given a distribution $P \in \Delta(X)$, the entropy is defined as:

$$\text{Entropy}(P) = -\sum_x P(x) \cdot \ln P(x)$$

Higher entropy means more uncertainty, for instance:

Uniform distribution has the highest entropy,
i.e., $\text{Entropy}(U(X)) = -\sum_x (1/|X|)\ln(1/|X|) = \ln(|X|)$

Deterministic distribution has zero entropy:
i.e., $\text{Entropy}(\delta(x_0)) = -1 \cdot \ln 1 - \sum_{x \neq x_0} 0 \ln 0 = 0$

Detour: Principle of Maximum Entropy

We want to find a distribution whose mean and covariance matrix equal to μ, Σ,
but there are infinitely many such distributions…

Detour: Principle of Maximum Entropy

We want to find a distribution whose mean and covariance matrix equal to μ, Σ,
but there are infinitely many such distributions…

Principle of Maximum Entropy:
Entropy Maximization subject to Moment Matching constraints

Detour: Principle of Maximum Entropy

We want to find a distribution whose mean and covariance matrix equal to μ, Σ,
but there are infinitely many such distributions…

Principle of Maximum Entropy:
Entropy Maximization subject to Moment Matching constraints

$$\max_{P \in \Delta(X)} \text{entropy}(P), \quad \text{s.t.,} \quad \mathbb{E}_{x \sim P}[x] = \mu, \quad \mathbb{E}_{x \sim P}[xx^\top] = \Sigma + \mu\mu^\top$$

Detour: Principle of Maximum Entropy

We want to find a distribution whose mean and covariance matrix equal to μ, Σ,
but there are infinitely many such distributions…

Principle of Maximum Entropy:
Entropy Maximization subject to Moment Matching constraints

$$\max_{P \in \Delta(X)} \text{entropy}(P), \quad \text{s.t.,} \quad \mathbb{E}_{x \sim P}[x] = \mu, \quad \mathbb{E}_{x \sim P}[xx^\top] = \Sigma + \mu\mu^\top$$

Solution: $P^\star = \mathcal{N}(\mu, \Sigma)$
(proof: out of scope)

Detour: Principle of Maximum Entropy

In summary:

Maximum Entropy Principle says that:

Among the distributions that satisfy pre-defined constraints (mean & variance),
let’s pick the one that is the most uncertain
(uncertainty measured in entropy)

Plan for Today:

1. The principle of Maximum Entropy

2. Constrained Optimization

3. The Algorithm: Maximum Entropy Inverse RL

Constrained Optimization:

Consider the following constrained optimization problem:

$$\min_{x} f(x)$$

$$s.t.,\ g_1(x) = 0, \quad g_2(x) = 0$$

Constrained Optimization:

Consider the following constrained optimization problem:

$$\min_{x} f(x)$$

$$s.t., g_1(x) = 0, \quad g_2(x) = 0$$

Denote $x^\star$ as the optimal solution here.

Constrained Optimization:

Consider the following constrained optimization problem:

$$\min_{x} f(x)$$

$$s.t., g_1(x) = 0, \quad g_2(x) = 0$$

Denote $x^\star$ as the optimal solution here.

How to solve such constrained optimization problem?

Constrained Optimization:

Define two Lagrange multiplier $w_1, w_2 \in \mathbb{R}$, we consider the following Lagrange formulation:

$$\min_{x} \left[\max_{w_1, w_2} f(x) + w_1 g_1(x) + w_2 g_2(x) \right]$$

Constrained Optimization:

Define two Lagrange multiplier $w_1, w_2 \in \mathbb{R}$, we consider the following Lagrange formulation:

$$\min_{x} \left[\max_{w_1, w_2} f(x) + w_1 g_1(x) + w_2 g_2(x) \right]$$

For any x that does not satisfy constraints, i.e., $g_1(x) \neq 0$ or $g_2(x) \neq 0$,
we must have: $\max_{w_1, w_2} f(x) + w_1 g_1(x) + w_2 g_2(x) = +\infty$

Constrained Optimization:

Define two Lagrange multiplier $w_1, w_2 \in \mathbb{R}$, we consider the following Lagrange formulation:

$$\min_{x} \left[\max_{w_1, w_2} f(x) + w_1 g_1(x) + w_2 g_2(x) \right]$$

For any x that does not satisfy constraints, i.e., $g_1(x) \neq 0$ or $g_2(x) \neq 0$,
we must have: $\max_{w_1, w_2} f(x) + w_1 g_1(x) + w_2 g_2(x) = +\infty$

For any x that satisfies constraints, i.e., $g_1(x) = 0$ and $g_2(x) = 0$,
we must have: $\max_{w_1, w_2} f(x) + w_1 g_1(x) + w_2 g_2(x) = f(x)$

Constrained Optimization:

Define two Lagrange multiplier $w_1, w_2 \in \mathbb{R}$, we consider the following Lagrange formulation:

$$\min_{x} \left[\max_{w_1, w_2} f(x) + w_1 g_1(x) + w_2 g_2(x) \right]$$

In other words,

$$\max_{w_1, w_2} f(x) + w_1 g_1(x) + w_2 g_2(x) = \begin{cases} +\infty & g_1(x) \neq 0 \text{ or } g_2(x) \neq 0 \text{ i.e, infeasible} \\ f(x) & g_1(x) = g_2(x) = 0 \text{ i.e., feasible} \end{cases}$$

Constrained Optimization:

Define two Lagrange multiplier $w_1, w_2 \in \mathbb{R}$, we consider the following Lagrange formulation:

$$\min_x \left[\max_{w_1, w_2} f(x) + w_1 g_1(x) + w_2 g_2(x) \right]$$

In other words,

$$\max_{w_1, w_2} f(x) + w_1 g_1(x) + w_2 g_2(x) = \begin{cases} +\infty & g_1(x) \neq 0 \text{ or } g_2(x) \neq 0 \text{ i.e, infeasible} \\ f(x) & g_1(x) = g_2(x) = 0 \text{ i.e., feasible} \end{cases}$$

Thus, solving the Lagrange formulation is equivalent to the original formulation:

$$\arg\min_x \left[\max_{w_1, w_2} f(x) + w_1 g_1(x) + w_2 g_2(x) \right] = x^\star$$

Constrained Optimization:

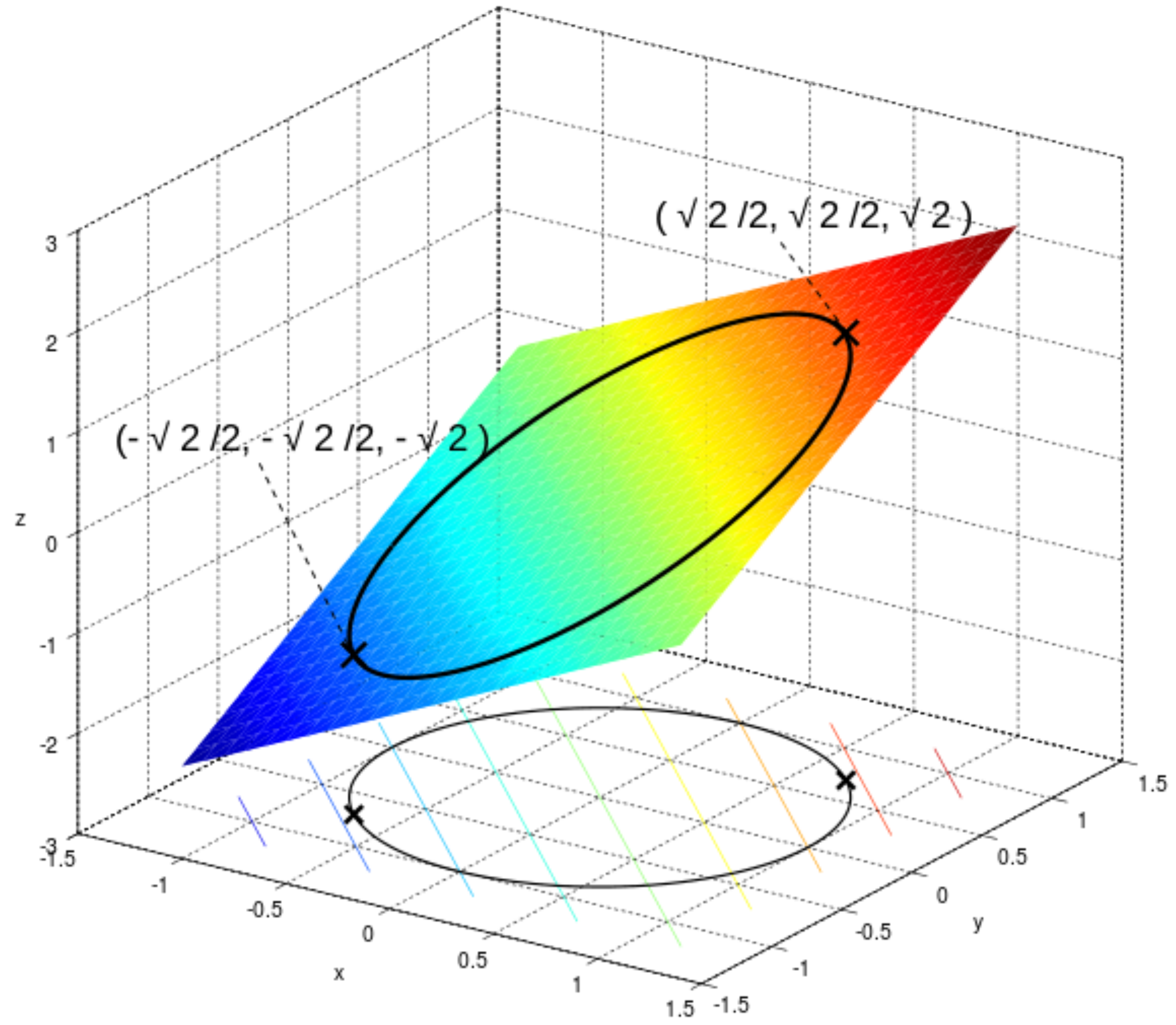
In summary, we have that

$$\arg\min_{x} \left[\max_{w_1, w_2} f(x) + w_1 g_1(x) + w_2 g_2(x) \right] = x^\star$$

Where $x^\star$ is the optimal solution of the original constrained program:

$$\min_{x} f(x)$$

$$s.t., g_1(x) = 0, \quad g_2(x) = 0$$



Example:

$$\min_{x,y} x + y, \text{ s.t.}, x^2 + y^2 = 1$$

Constrained Optimization:

We will often be interested in solving the dual version, i.e.,

$$\max_{w_1, w_2} \min_x \underbrace{f(x) + w_1 g_1(x) + w_2 g_2(x)}_{:= \ell(x,w)}$$

Constrained Optimization:

We will often be interested in solving the dual version, i.e.,

$$\max_{w_1, w_2} \min_x \underbrace{f(x) + w_1 g_1(x) + w_2 g_2(x)}_{:= \ell(x,w)}$$

And one procedure to solve a $\max\min$ **is the following iterative algorithm:**

Constrained Optimization:

We will often be interested in solving the dual version, i.e.,

$$\max_{w_1, w_2} \min_x \underbrace{f(x) + w_1 g_1(x) + w_2 g_2(x)}_{:= \ell(x, w)}$$

And one procedure to solve a $\max \min$ **is the following iterative algorithm:**

Initialize Lagrange multipliers w_1^0, w_2^0

Constrained Optimization:

We will often be interested in solving the dual version, i.e.,

$$\max_{w_1, w_2} \min_{x} \underbrace{f(x) + w_1 g_1(x) + w_2 g_2(x)}_{:= \ell(x, w)}$$

And one procedure to solve a $\max \min$ **is the following iterative algorithm:**

Initialize Lagrange multipliers w_1^0, w_2^0

For $t = 0 \rightarrow T - 1$

Constrained Optimization:

We will often be interested in solving the dual version, i.e.,

$$\max_{w_1,w_2} \min_x \underbrace{f(x) + w_1 g_1(x) + w_2 g_2(x)}_{:=\ell(x,w)}$$

And one procedure to solve a $\max \min$ **is the following iterative algorithm:**

Initialize Lagrange multipliers w_1^0, w_2^0

For $t = 0 \rightarrow T - 1$

$$x^t = \arg\min_x f(x) + w_1^t g_1(x) + w_2^t g_2(x) \quad \text{(\# best response: } \arg\min_x \ell(x, w^t))$$

Constrained Optimization:

We will often be interested in solving the dual version, i.e.,

$$\max_{w_1, w_2} \min_x \underbrace{f(x) + w_1 g_1(x) + w_2 g_2(x)}_{:= \ell(x, w)}$$

And one procedure to solve a $\max \min$ **is the following iterative algorithm:**

Initialize Lagrange multipliers w_1^0, w_2^0

For $t = 0 \to T - 1$

$$x^t = \arg\min_x f(x) + w_1^t g_1(x) + w_2^t g_2(x)$$ (# best response: $\arg\min_x \ell(x, w^t)$)

$$w_1^{t+1} = w_1^t + \eta g_1(x^t)$$

$$w_2^{t+1} = w_2^t + \eta g_2(x^t)$$

(#incremental update: $w^{t+1} = w^t + \eta \nabla_w \ell(x^t, w)$)

Constrained Optimization:

We will often be interested in solving the dual version, i.e.,

$$\max_{w_1, w_2} \min_x \underbrace{f(x) + w_1 g_1(x) + w_2 g_2(x)}_{:= \ell(x, w)}$$

And one procedure to solve a $\max \min$ **is the following iterative algorithm:**

Initialize Lagrange multipliers w_1^0, w_2^0

For $t = 0 \to T - 1$

$$x^t = \arg\min_x f(x) + w_1^t g_1(x) + w_2^t g_2(x)$$ (# best response: $\arg\min_x \ell(x, w^t)$)

$$w_1^{t+1} = w_1^t + \eta g_1(x^t)$$

$$w_2^{t+1} = w_2^t + \eta g_2(x^t)$$

(#incremental update: $w^{t+1} = w^t + \eta \nabla_w \ell(x^t, w)$)

Return: $\bar{x} = \sum_{t=0}^{T-1} x_t / T$

Constrained Optimization:

We will often be interested in solving the dual version, i.e.,

$$\max_{w_1,w_2} \min_x \underbrace{f(x) + w_1 g_1(x) + w_2 g_2(x)}_{:=\ell(x,w)}$$

And one procedure to solve a $\max \min$ **is the following iterative algorithm:**

Initialize Lagrange multipliers w_1^0, w_2^0

For $t = 0 \to T - 1$

$x^t = \arg\min_x f(x) + w_1^t g_1(x) + w_2^t g_2(x)$ (# best response: $\arg\min_x \ell(x, w^t)$)

$w_1^{t+1} = w_1^t + \eta g_1(x^t)$

(#incremental update: $w^{t+1} = w^t + \eta \nabla_w \ell(x^t, w)$)

$w_2^{t+1} = w_2^t + \eta g_2(x^t)$

Return: $\bar{x} = \sum_{t=0}^{T-1} x_t / T$ Informal theorem: when f, g are convex, $\bar{x} \to x^\star$, as $T \to \infty$

Plan for Today:

✓ 1. The principle of Maximum Entropy

✓ 2. Constrained Optimization

3. The Algorithm: Maximum Entropy Inverse RL

Maximum Entropy Inverse RL:

Maximum Entropy Inverse RL:

Q: we want to find a policy π such that $\mathbb{E}_{s,a\sim d_\mu^\pi}\phi(s,a) = \mathbb{E}_{s,a\sim d_\mu^{\pi^\star}}\phi(s,a)$

(Note **linear cost assumption** implies π is as good as $\pi^\star$)

But there are potentially many such policies…

Maximum Entropy Inverse RL:

Q: we want to find a policy π such that $\mathbb{E}_{s,a\sim d_\mu^\pi}\phi(s,a) = \mathbb{E}_{s,a\sim d_\mu^{\pi^\star}}\phi(s,a)$

(Note **linear cost assumption** implies π is as good as $\pi^\star$)

But there are potentially many such policies…

The principle of Maximum Entropy:

Find a policy π that maximizes some entropy while subject to the constraint:

Maximum Entropy Inverse RL:

Q: we want to find a policy π such that $\mathbb{E}_{s,a\sim d_\mu^\pi}\phi(s,a) = \mathbb{E}_{s,a\sim d_\mu^{\pi^\star}}\phi(s,a)$

(Note **linear cost assumption** implies π is as good as $\pi^\star$)

But there are potentially many such policies…

The principle of Maximum Entropy:

Find a policy π that maximizes some entropy while subject to the constraint:

$$\max_{\pi} \mathbb{E}_{s\sim d_\mu^\pi}\left[\text{entropy}\left(\pi(\cdot\,|\,s)\right)\right]$$

$$s.t, \mathbb{E}_{s,a\sim d_\mu^\pi}\phi(s,a) = \mathbb{E}_{s,a\sim d_\mu^{\pi^\star}}\phi(s,a)$$

Maximum Entropy Inverse RL:

Q: we want to find a policy π such that $\mathbb{E}_{s,a \sim d_\mu^\pi} \phi(s,a) = \mathbb{E}_{s,a \sim d_\mu^{\pi^\star}} \phi(s,a)$

(Note **linear cost assumption** implies π is as good as $\pi^\star$)

But there are potentially many such policies…

The principle of Maximum Entropy:

Find a policy π that maximizes some entropy while subject to the constraint:

$$\max_{\pi} \mathbb{E}_{s \sim d_\mu^\pi} \left[\text{entropy} \left(\pi(\cdot \mid s) \right) \right]$$

$$s.t, \mathbb{E}_{s,a \sim d_\mu^\pi} \phi(s,a) = \mathbb{E}_{s,a \sim d_\mu^{\pi^\star}} \phi(s,a)$$

This can be estimated using expert data: $\sum_{i=1}^{N} \phi(s_i^\star, a_i^\star)/N$

Maximum Entropy Inverse RL:

Let's simplify the objective $\max_\pi \mathbb{E}_{s\sim d_\mu^\pi}\left[\text{entropy}(\pi(\cdot \mid s))\right]$:

Maximum Entropy Inverse RL:

Let's simplify the objective $\max_{\pi} \mathbb{E}_{s\sim d_\mu^\pi}\left[\text{entropy}(\pi(\cdot\,|\,s))\right]$:

$$\mathbb{E}_{s\sim d_\mu^\pi}\left[\text{entropy}(\pi(\cdot\,|\,s))\right] = -\,\mathbb{E}_{s\sim d_\mu^\pi}\mathbb{E}_{a\sim\pi(\cdot|s)}\ln\pi(a\,|\,s) = -\,\mathbb{E}_{s,a\sim d_\mu^\pi}\ln\pi(a\,|\,s)$$

Maximum Entropy Inverse RL:

Let's simplify the objective $\max_{\pi} \mathbb{E}_{s \sim d_\mu^\pi} \left[\text{entropy}(\pi(\cdot \mid s)) \right]$:

$$\mathbb{E}_{s \sim d_\mu^\pi} \left[\text{entropy}(\pi(\cdot \mid s)) \right] = -\mathbb{E}_{s \sim d_\mu^\pi} \mathbb{E}_{a \sim \pi(\cdot \mid s)} \ln \pi(a \mid s) = -\mathbb{E}_{s,a \sim d_\mu^\pi} \ln \pi(a \mid s)$$

$$\arg\max_{\pi} \mathbb{E}_{s \sim d_\mu^\pi} \left[\text{entropy}(\pi(\cdot \mid s)) \right] = \arg\min_{\pi} \mathbb{E}_{s,a \sim d_\mu^\pi} \ln \pi(a \mid s)$$

Maximum Entropy Inverse RL:

We arrive at the following constraint optimization problem:

$$\arg\min_{\pi} \mathbb{E}_{s,a\sim d_\mu^\pi} \ln \pi(a\,|\,s)$$

$$s.t, \mathbb{E}_{s,a\sim d_\mu^\pi}\phi(s,a) = \mathbb{E}_{s,a\sim d_\mu^{\pi^\star}}\phi(s,a)$$

Introduce the Lagrange multiplier $w \in \mathbb{R}^d$ (we have d many constraints), consider the max-min dual version:

$$\max_{w\in\mathbb{R}^d} \min_{\pi} \mathbb{E}_{s,a\sim d_\mu^\pi} \ln \pi(a\,|\,s) + w^\top \left(\mathbb{E}_{s,a\sim d_\mu^\pi}\phi(s,a) - \mathbb{E}_{s,a\sim d_\mu^{\pi^\star}}\phi(s,a) \right)$$

Maximum Entropy Inverse RL:

Introduce the Lagrange multiplier $w \in \mathbb{R}^d$ (we have d many constraints), consider the max-min dual version:

$$\max_{w \in \mathbb{R}^d} \min_{\pi} \mathbb{E}_{s,a \sim d_\mu^\pi} \ln \pi(a \mid s) + w^\top \left(\mathbb{E}_{s,a \sim d_\mu^\pi} \phi(s,a) - \mathbb{E}_{s,a \sim d_\mu^{\pi^\star}} \phi(s,a) \right)$$

Next lecture,
we will design algorithm (in high level, it is the iterative algorithm framework)
for this $\max - \min$ problem