Distribution Matching, Maximum Entropy, GANs, and all that

Sanjiban Choudhury
Imitation Learning is NOT blindly copying the expert’s actions.
The Distribution Matching Problem

\[ P_{\text{expert}}(\xi^h) \]

(Unknown) expert distribution

\[ P_{\theta}(\xi) \]

Learn distribution over trajectories

All we see are expert samples

Learner can also generate samples

What loss should we use?
KL Divergence: A common measure!

Given two distributions $P(x)$ and $Q(x)$

$$D_{KL}(P || Q) = \sum_x P(x) \log \frac{P(x)}{Q(x)}$$
KL Divergence: A common measure!

\[ D_{KL}(P_{\text{expert}} \mid \mid P_{\theta}) = \sum_{\xi} P_{\text{expert}}(\xi) \log \frac{P_{\text{expert}}(\xi)}{P_{\theta}(\xi)} \]

Can we \( \min_{\theta} D_{KL}(P_{\text{expert}} \mid \mid P_{\theta}) \) if we don’t know \( P_{\text{expert}} \)?
KL Divergence: A common measure!

\[ P_{\text{expert}}(\xi^h) \]

(Unknown) expert distribution

\[ P_\theta(\xi) \]

Learn distribution over trajectories

Yes!

\[
\min_\theta D_{KL}(P_{\text{expert}} \mid \mid P_\theta) = \sum_\xi P_{\text{expert}}(\xi) \log \frac{P_{\text{expert}}(\xi)}{P_\theta(\xi)}
\]

\[
\min_\theta - \sum_\xi P_{\text{expert}}(\xi) \log P_\theta(\xi)
\]

Only need samples from expert!
Flying through a forest

Expert flies left and right of the tree

Given samples from expert
Flying through a forest

Expert flies left and right of the tree

Given samples from expert

Let’s say we want to learn $P_\theta(\xi)$, a gaussian over traj

$$\min_{\theta} D_{KL}(P_{\text{expert}}||P_\theta)$$

What will we learn?
Activity!
Think (30 sec): What Gaussian will we learn by minimizing KL divergence 

$$\min_{\theta} - \mathbb{E}_{\xi \sim P_{\text{expert}}(\xi)} \log P_{\theta}(\xi)$$

Pair: Find a partner

Share (45 sec): Partners exchange ideas
Forward KL is Mode-Covering!

- Makes sure probability is non-zero for every action the expert takes

  Maximizes recall

  But sacrifices precision, i.e. can leave expert support
Well what about Reverse KL?

\[
\min_\theta D_{KL}(P_\theta \mid \mid P_{\text{expert}})
\]

\[
\min_\theta \sum_\xi P_\theta(\xi) \log \frac{P_\theta(\xi)}{P_{\text{expert}}(\xi)}
\]

\[
\min_\theta - \sum_\xi P_\theta(\xi) \log P_{\text{expert}}(\xi) - H(P_\theta(.))
\]

Do we know this?

Entropy
Estimating Divergences
KL is part of a *spectrum* of divergences

f-divergence: A family of divergences

\[ D_f(P \mid \mid Q) = \sum_x Q(x) f \left( \frac{P(x)}{Q(x)} \right) \]

Where \( f() \) is a convex function

Ali and Silvey, 1966
KL is part of a spectrum of divergences

| Name               | $D_f(P||Q)$                                                                 | Generator $f(u)$                                    |
|--------------------|----------------------------------------------------------------------------|-----------------------------------------------------|
| Kullback-Leibler   | $\int p(x) \log \frac{p(x)}{q(x)} \, dx$                                | $u \log u$                                         |
| Reverse KL         | $\int q(x) \log \frac{q(x)}{p(x)} \, dx$                                | $- \log u$                                         |
| Pearson $\chi^2$  | $\int \frac{(q(x)-p(x))^2}{p(x)} \, dx$                                  | $(u - 1)^2$                                        |
| Squared Hellinger  | $\int \left( \sqrt{p(x)} - \sqrt{q(x)} \right)^2 \, dx$                | $(\sqrt{u} - 1)^2$                                |
| Jensen-Shannon     | $\frac{1}{2} \int p(x) \log \frac{2p(x)}{p(x)+q(x)} + q(x) \log \frac{2q(x)}{p(x)+q(x)} \, dx$ | $-(u + 1) \log \frac{1+u}{2} + u \log u$          |
| GAN                | $\int p(x) \log \frac{2p(x)}{p(x)+q(x)} + q(x) \log \frac{2q(x)}{p(x)+q(x)} \, dx - \log(4)$ | $u \log u - (u + 1) \log(u + 1)$                   |

Nowozin et al. 2017
Okay fine ... but how do we estimate these divergences when all we have are expert samples?
Use GANs to estimate divergence!

Minimize discriminator loss!

Maximally discriminate between learner and generator samples

θ

Generator

Discriminator

Nowozin et al. 2017
Use GANs to estimate divergence!

$$\min_{\theta} \max_{\phi} \mathbb{E}_{\xi \sim P_\theta(\xi)}[C_\phi(\xi)] - \mathbb{E}_{\xi \sim P_{\text{expert}}(\xi)}[f^*(C_\phi(\xi))]$$

Imitation Learning as \(\ell\)-Divergence Minimization

Liyiming Ke\(^1\), Saurabh Choudhury\(^1\), Mazr Raissi\(^1\), Wen Sun\(^1\), Gilwoo Lee\(^1\), and Siddhartha Srinivasa\(^1\)

\(^1\) Paul G. Allen School of Computer Science & Engineering, University of Washington, Seattle WA 98105, USA, \{lxyke, sajchou, mazr, wesuns, gilwoo, siddh\}@uw.edu

\(^2\) The Robotics Institute, Carnegie Mellon University, Pittsburgh PA 15213, USA, wenwu@andrew.cmu.edu
The Rise of Adversarial Imitation Learning

**JS-Divergence**

**Reverse-KL Divergence**

**Generative Adversarial Imitation Learning**

Jonathan Ho  
Stanford University  
hoj@cs.stanford.edu

Stefano Ermon  
Stanford University  
ermon@cs.stanford.edu

**Learning Robust Rewards with Adversarial Inverse Reinforcement Learning**

Justin Fu, Katie Luo, Sergey Levine  
Department of Electrical Engineering and Computer Science  
University of California, Berkeley  
Berkeley, CA 94720, USA  
justin@eecs.berkeley.edu, katielu8@berkeley.edu,  
slevine@eecs.berkeley.edu

**Jeffrey Divergence**

**State-Marginal f-divergence**

R2P2: A Reparameterized Pushforward Policy for Diverse, Precise Generative Path Forecasting

Nicolò Ruffinelli,  
Keepon Systems, Inc., USA  
neuroptica@gmail.com, jared.ruffinelli@gmail.com,  
and Paul Versace

Carnegie Mellon University, Pittsburgh PA 15213, USA  
neuroptica@gmail.com, jared.ruffinelli@gmail.com,  
and Paul Versace

Tianwei Ni, Harshil Sikchi, Yufei Wang, Tejas Gupta, Lisa Lee, Benjamin Eysenbach  
Carnegie Mellon University  
{tianwein, harshil, yufeiw2, tejiang, lslee, bseysenba}@cmu.edu
Which divergence do we care about?
What divergence do we care about?

f-divergence are great and all, but which one
do we actually care about?
What divergence do we care about?

What we actually care about is matching Performance Difference

$$J(\pi) = J(\pi^*)$$

$$\mathbb{E}_{\xi \sim P_\theta(\xi)} c(\xi) = \mathbb{E}_{\xi \sim P_{\text{expert}}(\xi)} c(\xi)$$

But we don’t know the costs $c(\cdot)$. 
What divergence do we care about?

What we actually care about is matching Performance Difference

\[ J(\pi) = J(\pi^*) \]

\[ \mathbb{E}_{\xi \sim P_{\theta}(\xi)} c(\xi) = \mathbb{E}_{\xi \sim P_{\text{expert}}(\xi)} c(\xi) \]

But we don’t know the costs \( c(.) \)

Costs are just weighted combination of features. What if we just matched all the expected features?
Proposal: Match cost features!

\[ P_{\text{expert}}(\xi^h) \]

(Unknown) expert distribution

\[ P_{\theta}(\xi) \]

Learn distribution over trajectories

All we see are expert samples

Learner can also generate samples
Proposal: Match cost features!

\[ P_{\text{expert}}(\xi^h) \]

(Unknown) expert distribution

\[ P_\theta(\xi) \]

Learn distribution over trajectories

All we see are expert samples

Learner can also generate samples

\[
\mathbb{E}_{\xi^h \sim P_{\text{expert}}} f_1(\xi^h) = \mathbb{E}_{\xi \sim P_\theta} f_1(\xi) \\
\mathbb{E}_{\xi^h \sim P_{\text{expert}}} f_2(\xi^h) = \mathbb{E}_{\xi \sim P_\theta} f_2(\xi) \\
\vdots \\
\mathbb{E}_{\xi^h \sim P_{\text{expert}}} f_k(\xi^h) = \mathbb{E}_{\xi \sim P_\theta} f_k(\xi)
\]
Let’s formalize!
Maximum Entropy Inverse Reinforcement Learning

Brian D. Ziebart, Andrew Maas, J. Andrew Bagnell, and Anind K. Dey
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
bziebart@cs.cmu.edu, amaas@andrew.cmu.edu, dbagnell@ri.cmu.edu, anind@cs.cmu.edu
Maximum Entropy Inverse Optimal Control

LEO: Learning Energy-based Models in Factor Graph Optimization

Paloma Sodhi\textsuperscript{1,2}, Eric Dexheimer\textsuperscript{1}, Mustafa Mukadam\textsuperscript{2}, Stuart Anderson\textsuperscript{2}, Michael Kaess\textsuperscript{1}
\textsuperscript{1}Carnegie Mellon University, \textsuperscript{2}Facebook AI Research
Maximum Entropy Inverse Optimal Control

Given dataset: \( \{ \xi_i^h, \phi_i \}_{i=1}^N \)  

(Sample dataset) (Map)

Solve for cost \( C_\theta(\xi) \)

\[
\min_{\theta} \frac{1}{N} \sum_{i=1}^N - \log P_\theta(\xi_i^h | \phi_i)
\]

Max lik. of human traj

\[
P_\theta(\xi | \phi) = \frac{1}{Z(\theta, \phi)} \exp(-C_\theta(\xi, \phi))
\]

More costly traj, less likely
for $i = 1, \ldots, N$

$$\xi_i \sim \frac{1}{Z} \exp \left( -C_\theta(\xi, \phi_i) \right)$$

$$\theta^+ = \theta - \eta \left[ \nabla_\theta C_\theta(\xi^h_i, \phi_i) - \nabla_\theta C_\theta(\xi_i, \phi_i) \right]$$

(Push down human cost)  
(Push up planner cost)
Maximum Entropy Inverse Optimal Control

\[
\text{for } i = 1, \ldots, N
\]

\[
\xi_i \sim \frac{1}{Z} \exp \left( -C_\theta(\xi, \phi_i) \right)
\]

\[
\theta^+ = \theta - \eta \left[ \nabla_\theta C_\theta(\xi_i^h, \phi_i) - \nabla_\theta C_\theta(\xi_i, \phi_i) \right]
\]

(Push down human cost) (Push up planner cost)

# Loop over datapoints
# Call planner!
# Update cost
for $i = 1, \ldots, N$

$$\xi_i \sim \frac{1}{Z} \exp \left( -C_\theta(\xi, \phi_i) \right)$$

$$\theta^+ = \theta - \eta \left[ \nabla_\theta C_\theta(\xi^h_i, \phi_i) - \nabla_\theta C_\theta(\xi_i, \phi_i) \right]$$

(Push down human cost)  (Push up planner cost)
for $i = 1, \ldots, N$

$$\xi_i \sim \frac{1}{Z} \exp \left( -C_{\theta}(\xi, \phi_i) \right)$$

$$\theta^+ = \theta - \eta \left[ \nabla_{\theta} C_{\theta}(\xi^h_i, \phi_i) - \nabla_{\theta} C_{\theta}(\xi_i, \phi_i) \right]$$

(Push down human cost)  
(Push up planner cost)
Deep Max Ent

Watch This: Scalable Cost-Function Learning for Path Planning in Urban Environments

Markus Wulfmeier¹, Dominic Zeng Wang¹ and Ingmar Posner¹
autonomous execution
1x real-time

goal

our method

IOC samples from $q(u_t | x_t)$
**Easy**

- Expert is realizable: $\pi^E \in \Pi$
- As $N \to \infty$, drive down $\epsilon = 0$ (or Bayes error)
- Nothing special. Collect lots of data and do Behavior Cloning

**Medium**

- Non-realizable expert but full expert support
- Even as $N \to \infty$, behavior cloning $O(\epsilon CT)$
  where $C$ is conc. coeff
- Requires interactive simulator (MaxEntIRL) to match distribution $\Rightarrow O(\epsilon T)$

**Hard**

- Non-realizable expert + limited expert support
- Even as $N \to \infty$, behavior cloning $O(\epsilon T^2)$
- Requires interactive expert (DAGGER / EIL) to provide labels $\Rightarrow O(\epsilon T)$
Learning to Search (LEARCH)

for $i = 1, \ldots, N$

$\xi_i^* = \min_\xi [C_\theta(\xi, \phi_i) - \gamma(\xi, \xi^h)]$

$\theta^+ = \theta - \eta [\nabla_\theta C_\theta(\xi_i^h, \phi_i) - \nabla_\theta C_\theta(\xi_i^*, \phi_i) + \nabla_\theta R(\theta)]$

(Maximum Entropy Inverse Optimal Control)

When the expert is
Suboptimal
Noisy
Privileged Information

LEARCH does NOT converge!!