1) Imitation Learning from observations by minimizing inverse dynamics disagreement

General Notes

→ minimizing the $\boxed{\text{discrepancy in the inverse dynamics models}}$ of expert and agent.

→ model free

→ upper bound : negative entropy of state action occupancy measure

→ Different occupancy measures for MDP :-

i) State Action

ii) State Transition

iii) Joint Occupancy

→ Agent and expert share $\boxed{\text{same dynamics}}$.

→ $\boxed{\text{without access of dynamics model}}$ and expert guidance.

2) To Follow or Not to Follow

General Notes

→ learns to follow demonstrations by $\boxed{\text{aligning}}$ the timescale and $\boxed{\text{skipping infeasible parts}}$.

→ $\boxed{\text{Hierarchy}}$ of policies :- i) $\boxed{\text{Meta}}$ → Sub Goal

ii) $\boxed{\text{Low-Level}}$ → How to get to

→ currents observations and demonstrations are collected by different agents in different $\boxed{\text{environments}}$, thus embeddings are used.

Date
Page

3) IL by State Only Distribution Matching

General Notes

→ AIL methods are often ustable & lack a reliable convergence estimator.
→ This paper proposes a non adversarial LfO approach.
→ Training objective minimizes the KLD
→ Estimates the expert state transition distribution using normalizing flows (trained offline)
→ To match transition distributions of policy and expert exactly, the state-next-state distribution of policy is expanded into policy entropy, forward dynamics and inverse action model

4) Decoupled Policy Optimization

General Notes

→ Decouples policy as a high level state planner and an inverse dynamics model.
→ Uses embedded decoupled policy gradient with GAN.
→ Takes a supervised approach, but has GAT

5) Differentiable Physics

→ Incorporates the differential physics simulator as a physics prior into its computational graph for policy learning.
→ ILD unrolls dynamics by sampling actions from a parameterized policy and minimizing distance.
→ Selects learning objectives for each state dynamically.

6) Raw Video via Context Translation

General Notes

→ Context translation + deep RL

→ Deals with domain shift

7) Self Supervised Adversarial IL

→ By incorporating a discriminator

i) Disposes manual intervention

ii) Guiding function approximation based on the state transition of expert trajectories

iii) Avoids a no action performance

8) MOBILE

→ Uses the distribution matching IL framework.

① Imitation from Observation

$p(o_t | s_t, \omega)$; $p(s_{t+1} | s_t, a_t, \omega)$; $p(a_t | s_t, \omega)$

$\omega \rightarrow$ context

Further challenge: Domain shift

~~[illegible]~~ 2 problems: $\rightarrow$ i) What info from observations to track in own context.

ii) Actions to track demonstrated observations

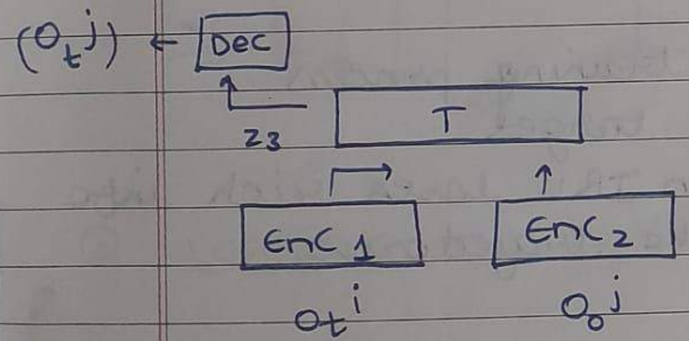
$D_i = [o_0^i, o_1^i, \ldots o_T^i] \rightarrow$ source context

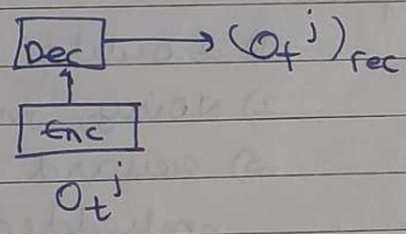
$D_j = [o_0^j, o_1^j, \ldots o_T^j] \longrightarrow$ target context

$\rightarrow$ "The model first learns to output observations in D_j conditioned on D_i and first obs o_0^j and context ω_j.

$\rightarrow$ Assume D_i, D_j are aligned in time.

$$M(o_t^i, o_0^j) = (o_T^j)_{trans}$$





$$L_{Trans} = \| (\hat{o}_t^j)_{trans} - o_t^j \|_2^2$$

$$L_{rec} = \| Dec(Enc(o_t^j)) - o_t^j \|_2^2$$

$$L_{align} = \| z_3 - Enc(o_t^j) \|_2^2$$

$L = L_t + \lambda_1 L_r + \lambda_2 L_a$

hyperparams

→ Reward Func^n's for Tracking Features.

$$\hat{R}_{feat}(o_t^l) = -\| Enc_1(o_t^l) - \frac{1}{n}\sum F(o_t^i, o_0^l)\|_2^2$$

(*) Distribution of observations fed into Enc_1 may not be same during policy learning and training. We need a reward that directly penalizes the policy for exp observations that differ from translated obs using M :-

$$\hat{R}_{imp}(o_t^l) = -\| o_t^l - \frac{1}{n}\sum M(o_t^i, o_0^l)\|_2^2$$

② Imitation Learning as State Matching via Differentiable Physics

- AIL and IRL have three main limitations due to learning an additional intermediate signal :→
  1) Double loop training process
  2) Noisy, moving target
  3) reward func in IRL loses rich info embedded in the trajectories.

$$x_{t+1} = x_t + \Delta t\, v_t\ , \quad v_{t+1} = v_t + \frac{f\,\Delta t}{m}$$

→ π_θ is used to unroll → trajectory τ_θ → $S_{0:H}$, → $A_{0:H}$

$T(s_{t+1} | s_t, a_t)$ is our evn. dynamics.

$$s_{1:H} = \alpha(s_0, a_{0:H})$$

$$\arg\min_{\theta} \sum_{s \sim \tau_\theta} \sum_{t=0}^{T} (g_t - s_t)^2$$

$\mathcal{L}_{triplet}$ ⇒ enforces encodings sim of adj frames

$\mathcal{L}_{ae}$ ⇒ permits enc to be dec to image

$s = g_\theta(O)$, $s_p = g_\theta(O_p)$

$$\mathcal{L}_{ae}(\theta, \gamma) = \| O - q_\gamma(g_\theta(O)) \|^2$$

Finally,

$$\{R, G, B\} \rightarrow \{L, ab\}$$

$$s = g(\theta) = [g_{\theta_1}(v_1), g_{\theta_2}(v_2)]$$

$$\mathcal{L}_{frame}(\theta, \gamma) = L_s(\theta) + L_{triplet}(\theta) + L_{ae}(\theta, \gamma)$$

③ Sequence encoding

i) L_2 (DPC)

$$D = \{O^i\}_{i \leq N} = \{O^i_{0:T}\}_{1 \leq i \leq N}$$ → Dataset of D videos

$$\hat{s}_{t+1} = d_\phi(f_w(s_1 \ldots s_t))$$

$$\hat{s}_{t+k} = d_\phi(f_w(s_1 \ldots \hat{s}_{t+k-1}))$$

$$L_2(\theta, w, \phi) = -\mathbb{E}_{O_{0:T}}\left[\frac{1}{K}\sum_{1 \leq k \leq K} L_{t+k}\right]$$

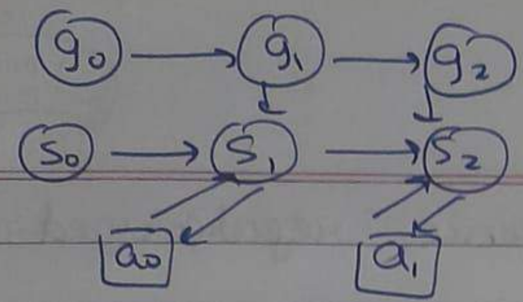
$$L_t = \log \frac{h(\hat{s}_t, s_t)}{h(\hat{s}_t, s_t) + \sum_{t \neq t'} h(\hat{s}_t, s_t)}$$

where $h(a,b) = \exp\left(\frac{a^T b}{\tau \|a\| \|b\|}\right)$ τ ⇒ temp

- L_0 ⇒ bootstrapped expert behaviour loss, which encourages separation between expert and non expert trajectories

$$L_0(\theta, w) = -\mathbb{E} \log \frac{h(z, z_p)}{h(z, z_p) + \sum_i h(z, z_{n,i})}$$

$$z = f_w(g_\theta(O_0) \ldots g_\theta(O_T))$$



We use Chamfer-α Loss $\rightarrow$ Deviation Loss + Coverage Loss

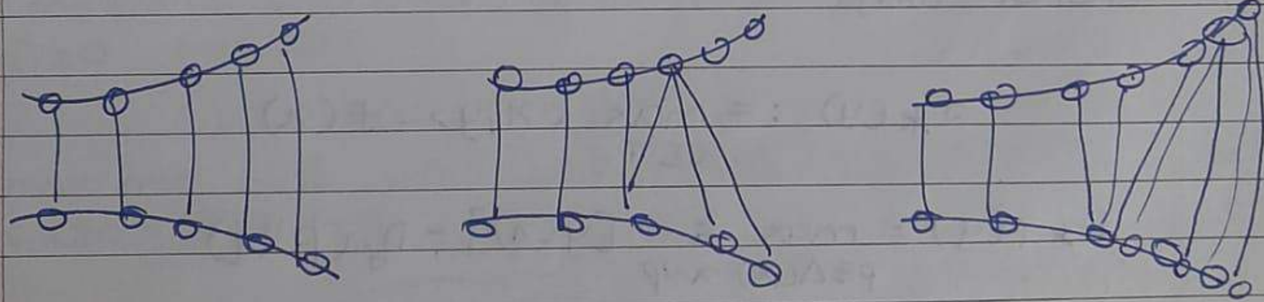
$$L_d = \frac{1}{|T_\theta|} \sum_{s_t \in T_\theta} \min_{g \in T_{exp}} \|g - s_t\|_2^2$$

$\Rightarrow$ But, this may lead to state collapse.

eg. Consider a large number of states close to small subset of T_{exp}.

$$L_g = \frac{1}{|T_{exp}|} \sum_{g_t \in T_{exp}} \min_{s_t \in T_\pi} \|g_t - s\|_2^2$$

$$L_{Chf-\alpha} = L_d + \alpha L_g$$



③ LOBSDice $\rightarrow$ (verify for SOIL)

④ Versatile Offline Imitation from observations...

High Level :- $\min_\pi D_{KL}(d^\pi(s) \| d^E(s))$

- Naively optimizing the state occupancy distribution would result in an actor critic style IL algorithm.

$$= \min_{\pi} - E_{(s,a) \sim d^{o}} [\, \mathcal{E}^{*}(s,a) \log \pi(a|s)]$$

where $\mathcal{E}^{*}(s,a) = \dfrac{d^{*}(s,a)}{d^{o}(s,a)}$

(5) Imitation from observation with Bootstrapped CL

→ 2 main approaches for IfO are - reward methods
- adverserial methods

Advantages / Motivation behind this approach.

(i) To learn and estimate at each timestep the system state using agent's visual obs.

(ii) Identify behaviour induced by agent and make sure it serves same purpose as expert

(1) Agent Training
- 2 phases → Alignment / Interactive

(i) Alignment

$$O \sim p(D_e)\ ,\ O \sim p(D_a)$$

$f_w \Rightarrow$ sequence encoding function.
$g_\theta \Rightarrow$ image encoding function

(ii) Interactive

PRQ-V2 ← → RL task where reward = distance(O_e, O_a)

(4). Sample an expert episode

$$O_e \sim p(D_e)$$

$$d(O_{0:t}, O_{e\,0:t}) = \| f_w(g_\theta(O_{0:t})) - f_w(g_\theta(O_{e\,0:t}))\|$$

$$r_t = -d(O_{0:t}, O_{e\,0:t})$$

continue to train f_w, g_θ on new trajectories

(2) Image Encoding

$$\mathcal{L}_{triplet}(\theta) = \|s - s_p\|^2 + \max(p - \|s - s_n\|^2, 0)$$

→ To bring sequences from the same distribut
close together we use L_O.
For O belonging to one sequence,
$O_p \Rightarrow$ same distribution
$O_{n_i} \Rightarrow$ different distribution

$$\mathcal{L}_{seq}(\theta, w, \phi) = L_Z(\theta, w, \phi) + L_O(\theta, w)$$

⑥ Policy Contrastive Imitation Learning

→ Problem :- AIL → Low quality of discriminator representation, since it is trained via binary classification.
→ <u>Use contrastive learning loss</u>
- Push the experts representation together and pull the agents away from them.

⇒ <u>CPIL</u>
→ Common heuristics dictate that x_p is just an augmentation to x, but this is not strong consideration ~~point~~ for AIL, since we need to discern good/b behaviour.
→ Good and bad states may look similar or the difference can be minor.
→ This causes a diff b/w positive sample and an sample; overwhelms the diff b/w good / bad.

$$\mathcal{L} = \mathbb{E}\left[\phi(x_0)^T \phi(x_p) + \log(\exp \phi(x_0)^T \phi(x_p) + \sum_i \exp \phi(x_0)^T \phi(x_i)\right]$$

$$r(x) = \phi(x)^T \mathbb{E}_{x \in D} \phi(x_E)$$

Page

(7) versatile skill control via self-supervised

→ Given a large dataset of unlabelled motion clips with diverse behaviours, extracting and learning individual sensible skills can be challenging.

→ One promising attempt for unsupervised skill discovery is based on maximizing the discriminability of skills represented by latent variables on which policy is conditioned.

→ Policy training signals are often constructed from variational approximations of the $\boxed{\text{mutual info between}}$ latent variables and traversed state histories using a learned skill discriminator.

EXISTING ⇒ 1) Collect disparate reference motions and learn individual skills separately.
2) Stack training discriminators over separate tasks

- APPROACH :→ i) imitation discriminator (d_ψ)
  ii) skill discriminator (q_ϕ)

$$o_t^I = (o^I_{t-H^I+1} \dots o_t^I)$$

$$s_t^I = (s_{t-H+1} \dots s_t)$$

1) Imitation Discriminator Formulation

$$\mathbb{E}_{d^M}\left[(d_\psi(o^I)-1)^2\right] + \mathbb{E}_{d^\pi}\left[(d_\psi(f^I(s))+1)^2\right]$$

$$r^I = \max\left[0,\ 1-0.25(d_\psi(f^I(s))-1)^2\right]$$

$$r \in [0,1]$$

- Introducing f-divergence regularized state matching objective.
- Using dual optimality solution to formulate a weighted regression policy objective that amounts to BC of optimal policy

// Prelim $d^{\pi}(s,a) := (1-\gamma)\sum_t \gamma^t Pr(s_t=s, a_t=a)$

$$d^{\pi}(s,a) = (1-\gamma)\mu_0(s)\pi(a|s) + \gamma . T_*^{\pi} d^{\pi}(s,a)$$

$$T_* d^{\pi}(s,a) := \pi(a|s) \sum T(s|\tilde{s},\tilde{a}) d(\tilde{s},\tilde{a})$$

i) f-Divergence

$$D_f(p||q) = \mathbb{E}_{x\sim q}\left[f\left(\frac{p(x)}{q(x)}\right)\right]$$

ii) Fenchel conj

$$f_*(y) := \max_{x\in\pi} \langle x,y\rangle - f(x)$$

$$D_* f(y) = \max_{p\in\Delta(X)} \mathbb{E}_{x\sim p}[y(x)] - D_f(p||q)$$

$$= \mathbb{E}_{x\sim q}[f_*(y(x))]$$

// Algo Assumption-1 $d^{o}(s) > 0$ when $d^{E}(s) > 0$

↳ coverage

Assumption-2

$$D_{KL}(d^{\pi}(s)||d^{E}(s)) \leq$$

$$\mathbb{E}_{s\sim d^{\pi}}\left[\log\left(\frac{d^{o}(s)}{d^{E}(s)}\right)\right] + D_{KL}(d^{\pi}(s,a)||d^{o}(s,a))$$

Any f-divergence $D_f \geq D_{KL}$,

f-divergence regularized state occupancy matching objective

$$D_{KL}(d^{\pi}(s) \| d^{E}(s)) \leq \underline{\mathbb{E}_{s\sim d^{\pi}}\left[\log\left(\frac{d^{0}(s)}{d^{E}(s)}\right)\right] + D_f(d^{\pi}(s,a) \| d^{0}(s,a))}$$

① Discriminator Training.

$$\min_{c} \mathbb{E}_{s\sim d^{E}}[\log c(s)] + \mathbb{E}_{s\sim d^{0}}[\log(1-c(s))]$$

$$c^{*}(s) = \frac{d^{0}(s)}{d^{0}(s)+d^{E}(s)}$$

② Dual Value Func^n Training

$$\max_{d(s,a)\geq 0} \mathbb{E}_{s\sim d(s,a)}[R(s)] - D_f(d \| d^{0})$$

$$\max_{d(s,a)\geq 0}\min_{V(s)\geq 0} \mathbb{E}_{s\sim d}[R(s)] - D_f(d\|d^{0}) + \sum_{s} V(s)\left((1-\gamma)\mu_0(s) + \gamma T_{*} d(s) - \sum d(s,a)\right)$$

$$\underline{\sum_{s} V(s)\cdot T_{*} d(s)} = \sum d(s,a)\cdot TV(s,a)$$

$$\Rightarrow \min_{V(s)\geq 0}\max_{d(s,a)\geq 0} (1-\gamma)\mathbb{E}_{s\sim\mu_0}[V(s)] + \mathbb{E}_{(s,a)\sim d}[(R(s)+\gamma TV(s,a)-V(s))] - D_f(d(s,a)\|d^{0}(s,a))$$

$$\Rightarrow \min_{V(s)\geq 0} (1-\gamma)\mathbb{E}_{s\sim\mu_0}[V(s)] + \mathbb{E}_{(s,a)\sim d^{0}}[f_{*}(R(s)+\gamma TV(s,a)-V(s))]$$

③ Weighted Regression Policy Training

$$\min_{\pi} -\mathbb{E}_{(s,a)\sim d^{*}}[\log \pi(a|s)]$$

ILPO

① Assumes actions as discrete.

↳ $\{s_1^*, s_2^* \ldots s_N^*\} \in D$. → Predict a forward model.

Predict next state $s_{t+1}, \Leftarrow G_\theta(E_p(s_t), z)$

embedding

↳ $L_{min} = \min_{z} \|\Delta_t - G_\theta(E_p(s_t), z)\|^2$.

↳ $\boxed{\text{imp}}$ penalize closest to true next observation

② Latent Policy $\pi_\omega(z \mid E_p(s_t))$

→ $\hat{s}_{t+1} = \mathbb{E}_{\pi_\omega}[s_{t+1} \mid s_t]$

$= \sum G_\theta \cdot \pi_\omega$ → $L_{exp} = \|s_{t+1} - \hat{s}_{t+1}\|^2$

→ $L_{net} = L_{exp} + L_{min}$.

③ Aligning Actions

↳ collect $\{s_t, a_t, s_{t+1}\}$ → Proceed in supervised manner.

use G for z_t → (softmax).

$$J(\pi_\theta) = \sum_{t=0}^{T} \mathbb{E}_{(s_t,a_t)\sim \mu^{\pi_\theta}} \left[r(s_t,a_t) + \alpha H(\pi_\theta(\cdot|s)) \right]$$

- Look at Soft Actor Critic Method.

Method

$$\mu^{\pi_\theta}(s_T, \ldots s_0) = p(s_0) \prod_{i=0\ldots T-1} \mu^{\pi_\theta}(s_{i+1}|s_i)$$

$$\mu^{E}(s_T, \ldots s_0) = p(s_0) \prod_{i=0\ldots T-1} \mu^{E}(s_{i+1}|s_i)$$

goal is to match state-only trajectory distribution μ^{π_θ} induced by the policy with the state only expert trajectory distribution μ^E by minimizing the KL between them.

$$J_{SOIL-TDM} = \mathbb{D}_{KL}(\mu^{\pi_\theta} \| \mu^E)$$

$$= \mathbb{E}_{(s_T,\ldots s_0)\sim \mu^{\pi_\theta}} [\log \mu^{\pi_\theta} - \log \mu^E]$$

$$= \sum_{i=0..T-1} \mathbb{E}_{(s_{i+1},s_i)\sim \mu^{\pi_\theta}} [\log \mu^{\pi_\theta}(s_{i+1}|s_i) - \log \mu^E(s_{i+1}|s_i)]$$

$$\mu^{\pi_\theta}(s_{i+1}|s_i) = \frac{p(s_{i+1}|a_i,s_i)\,\pi_\theta(a_i|s_i)}{\pi'_\theta(a_i|s_i,s_{i+1})}$$

$$\Rightarrow \min \sum_{i=0\ldots T-1} \mathbb{E}_{(s_i,a_i,s_{i+1})\sim \pi_\theta} \left[\log p(s_{i+1}|a_i,s_i) + \log \pi_\theta(a_i|s_i) - \log \pi'_\theta(a_i|s_{i+1},s_i) - \log \mu^E(s_{i+1}|s_i) \right]$$

$$r(a_i,s_i) := \mathbb{E}_{s_{i+1}\sim p(s_{i+1}|s_i,a_i)} \left[-\log p(s_{i+1}|a_i,s_i) + \log \pi'_\theta(a_i|s_i,s_{i+1}) + \log \mu^E(s_{i+1}|s_i) \right]$$

$$\min \mathbb{D}_{KL}(\mu^{\pi_\theta} \| \mu^E)$$

$$= \max \sum_{i=0..T-1} \mathbb{E}_{(a_i,s_i)\sim \pi_\theta} \left[-\log \pi_\theta(a_i|s_i) + r(a_i,s_i) \right]$$

$$= \max \sum_{i=0\ldots T-1} \mathbb{E}_{(a_i,s_i)\sim \pi_\theta} \left[r(a_i,s_i) + H(\pi_\theta(\cdot|s)) \right]$$

Q) Reverse dynamics model how to code it?

.. Bayes formulation

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}$$

$$P(s_{i+1}|s_i) = \frac{P(s_i|s_{i+1})\,P(s_{i+1})}{P(s_i)}$$

$\cancel{P(s_{i+1}) = P(s_{i+1}|s_i)}$ $\qquad \pi(a_i|s_i)\,p(s_{i+1}|s_i,a_i)$

$p(s$

SILO Paper.

Naively following a demonstration does not work when demonstration consists a state that is unreachable by the learner agent.

- Uses a hierarchical RL framework

meta Policy → chooses subgoal o_g^T in demonstrations τ.

low-level Policy → Bring current state closer to subgoal o_g^T, terminated when close enough to o_g^T.

$\pi_{meta}(g \mid o_t, \tau; \theta)$

$o_t \Rightarrow$ current observation
$\tau \Rightarrow \{o_1^T, o_2^T, o_3^T \ldots o_T^T\} \Rightarrow$ demonstration to imitate
$g \in [T, T] \Rightarrow$ demonstration state index of sub goal.

Once meta policy chooses o_g^T as a sub goal, the goal conditioned low-level policy generates an action $a_t \sim \pi_{low}(a \mid o_t, o_g^T; \phi)$. $\Rightarrow$ generates a rollout

until $|o_g^T - o_{t+\Delta}| < \varepsilon$.

Reward for meta policy $\Rightarrow$ 1/0

(A) Uses embeddings to condense into common vector space.

SAIL (Self Supervised Adversarial imitation learning)

- 4 Models →
  - i) M) $P(\hat{a} | S_t, S_{t+1})$ → Inverse Dynamics Model
  - ii) π_θ ⇒ Policy Model
  - iii) G → Generative Model
  - iv) D → Discriminator Model

- Overview of Algorithm
  - i) Random weights initialization
  - ii) Use π_θ to collect samples as (S_t, a, S_{t+1}) (I_s)
  - iii) Train M, and predict pseudo labels $\hat{A}$ for T_e
  - iv) Train π_θ using BC Approach
  - v) Also updates G during training
  - vi) Use π_θ (updated) to create new samples for M ~~to learn~~ action T_e.
  - vii) Append to I_s all samples D cannot differentiate between J^e and $G(I^{pos})$

- Goal aware function

$$\min_{M \cup \pi} \max_{D} SAIL(M, \pi, G, D) = \mathbb{E}_{r \sim R(\pi_e, Env)} [\log (D(r))] + \mathbb{E}_{r' \sim R(\pi_\theta, Env)} [\log (1 - D(G(r', \pi_\theta(r'))))]$$

NOTE:- Beginning bad samples appended to I_s.
To avoid overfitting early, use replay buffer
$RB(S_t, \dots S_{t+n}) \sim T^e \cup T^\pi$.

- Generative Model
  - i) Used at Behaviour Cloning func^n.
  - ii) Selecting samples appended to I_s

→ SAIL allows G to update π_θ to create actions to correctly condition G to ~~update π_θ, create actions~~ generate correct state transitions, equal to those observed. } B1

→ G becomes a forward dynamics model.

2 Problems

M may not always be correct

M may be stuck at some local minima.

$$L_G = -\frac{1}{N} \left[\sum_{i=1}^{N} S_{i+1} \cdot \log (G(S_i, \pi_\theta(S_i))) \right]$$

B2) D directly updating weights ⇒ π_θ has direct-temporal signal of where it deviates.

① The task is formulated as →

$D = \{x_1^d, x_2^d \ldots\}$.

↳ i) At state x_0, take action $\pi(x_0, x_1^d, \theta) \to x_0'$.

If goal Recognizer(x_0', x_1^d) is High,

$x_0 = x_0'$, take action.

while loop. till low.

② Learning GSP is entropy loss → $L(a_t, \tilde{a}_t)$

$p(a_t)$ → actual

→ predicted action dis

↳ Problem → multimodality

③ Instead of penalizing actions, penalize closeness on next observed state from both distribution.

$$\rightarrow \min \|x_{t+1} - \tilde{x}_{t+1}\|^2 + \lambda \|x_{t+1} - \hat{x}_{t+1}\|^2 + L(a_t, \hat{a}_t)$$

↳ we need a good forward model 'f'.

State Alignment Based Imitation Learning

* 4 pointers
  - i) Inverse State Based Model ✓
  - ii) Deviation Correction (β-VAE)
  - iii) Global Alignment
  - iv) Regularized Policy Update

→ Local Alignment → Policy Prior
Global Alignment → Rewards

$\phi \Rightarrow$ discriminator

→ VAE to produce next state

$$W = E_{s \sim T_e}[\phi(s)] - E_{s \sim T}[\phi(s)].$$

$$r(s_i, s_{i+1}) = \frac{1}{T}\left(\phi(s_{i+1}) - E_{s \sim T_e}\phi(s)\right)$$

State Only Imitation Learning for Dexterous Manipulation

① similar to BCO, but joint iterative training.
and
②

$$a_t' = h_\phi(s_t, s_{t+1}) \, , \; L_{inv} = \|a_t' - a_t\|^2$$

to train inverse model.

② optimize $g_{soil} = g + \lambda_0 \lambda_1^k + \sum \nabla_\theta \log \pi_\theta(a'|s)$

Adversarial Imitation Learning from State Only Demonstrations

① Experts Data → Critic; Imitators Data → Actor

↳ D_θ, π_ϕ.

Takes action according to π_ϕ, τ.

→ actual trajectory

$$D_\theta \doteq -(\mathbb{E}_\tau [\log(D_\theta(s,s'))]\} + \mathbb{E}_{\tau_E}[\log(1-D_\theta(s,s'))].$$

π_ϕ → TRPO updates.

$\boxed{\text{BCO}}$

① ~~Use~~ Learn an Inverse Dynamics Model by first letting agent ~~of~~ explore pre-demonstration. (π_ϕ)

② we have 'D' which contains numerous trajectories. We use our inverse dynamics model on the agent specific part to infer action $\tilde{a}_i$.

③ ~~Now we try~~ Cast this problem as BC. → ~~get~~ with $(s_i, \hat{a}_i)$ pairs.

find θ st probability of $(\check{a}_i \cancel{\text{and}} | s_i)$ is max.