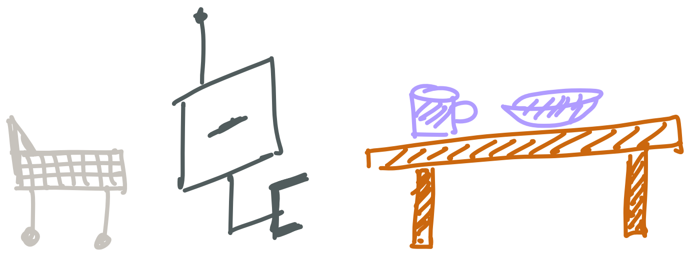
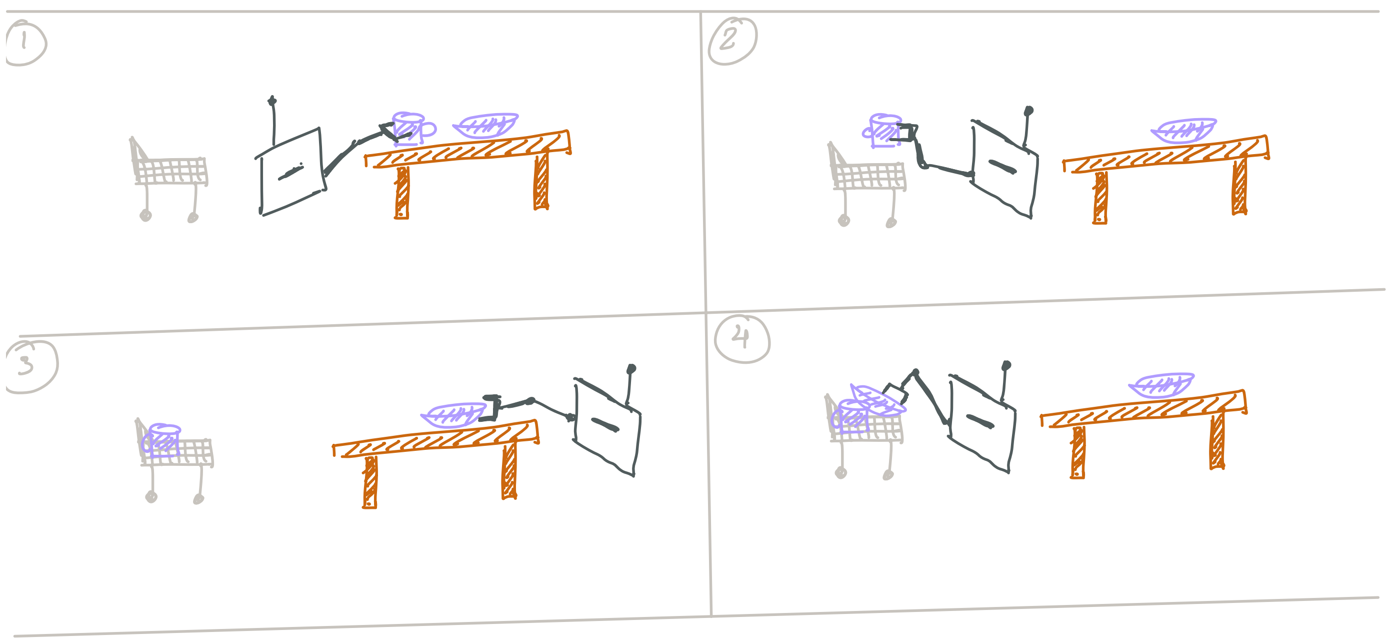




Formulate as a Markov Decision Process (MDP)

$$\langle S, A, R, T \rangle$$

$$s_1 \xrightarrow{a_1} s_2 \xrightarrow{a_2} s_3$$

S:

A:

R:

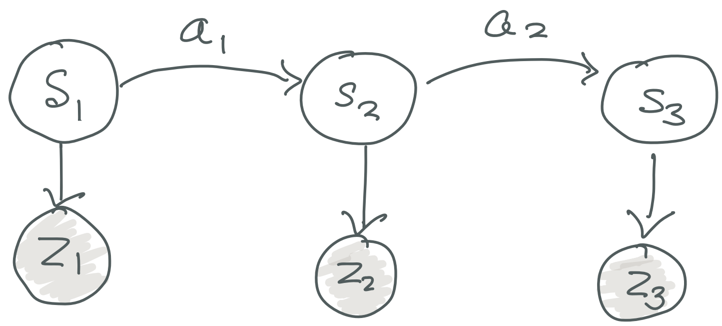
$T(s' | s, a)$:

PERCEPTION

How do we estimate State from Observation?

RGBD Image
(observation Z_t)

We don't see state,
we see observations



State

- Pose of Objects
- Configuration of Robot

Observation

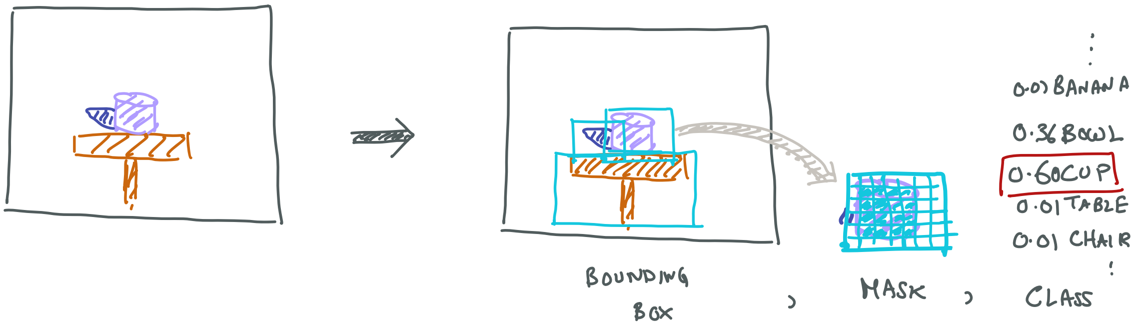
- RGBD Image
- Encoders on Joints
- Force-Torque Sensor
- IMU

HOW DO WE ESTIMATE 3D POSE OF OBJECTS FROM RGBD IMAGE ?

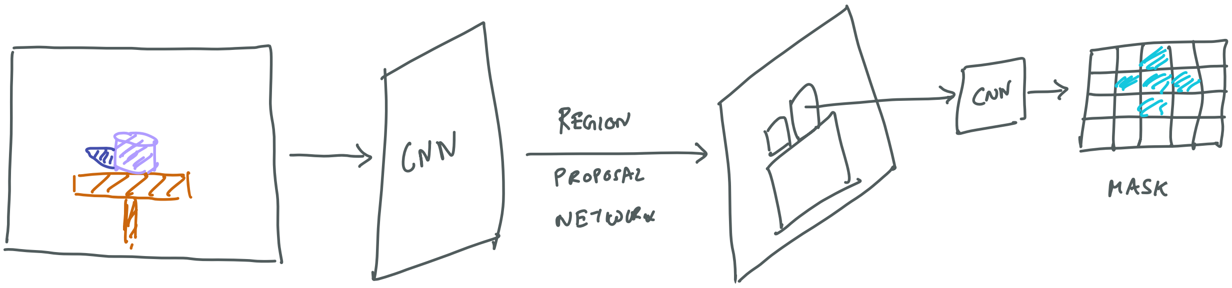
STEP 1

TRAIN AN INSTANCE SEGMENTATOR (MASK-RCNN)

TO PROPOSE : (BOUNDING BOX , MASK , OBJECT CLASS)

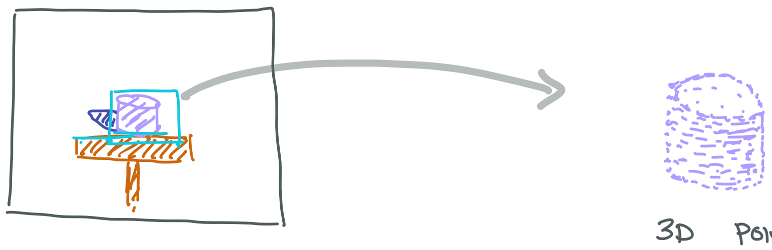


HOW ?



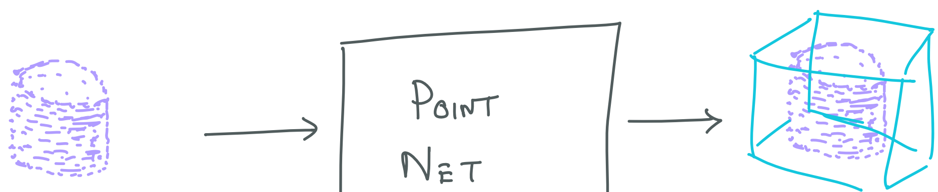
Step 2

Extract 3D Point Cloud from 2D image



Step 3

Train Point Net to predict accurate 6DoF pose from point clo



x, y, z

ϕ, θ, ψ

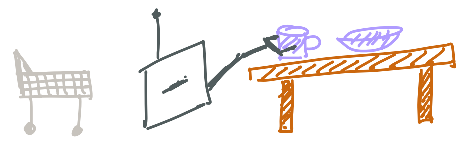
LET'S TAKE STOCK : WE KNOW STATE, WE KNOW ACTIONS

$$\langle S, A, R, T \rangle$$

GOAL:

LEARN A POLICY π THAT MAPS STATE TO ACTIONS.

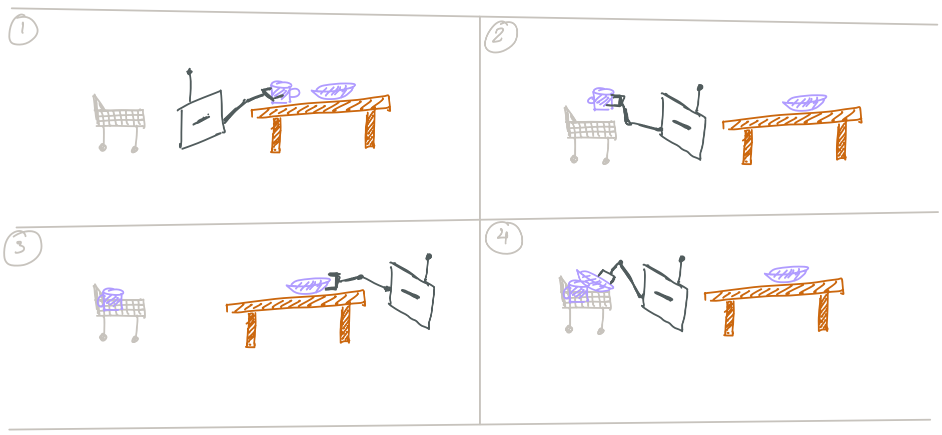
$$\pi : S \longrightarrow A$$



WHAT IS THE SIMPLEST APPROACH ?

BEHAVIOR CLONING

STEP 1: HUMAN PROVIDES DEMONSTRATIONS



$$D = \left\{ \begin{array}{l} (s_1^*, a_1^*, s_2^*, a_2^*, \ldots \ldots s_T^*, a_T^*) \\ (s_1^*, a_1^*, s_2^*, a_2^*, \ldots \ldots s_T^*, a_T^*) \\ (s_1^*, a_1^*, s_2^*, a_2^*, \ldots \ldots s_T^*, a_T^*) \end{array} \right\}$$

STEP 2: TRAIN A POLICY TO MAP STATES TO ACTIONS

$$\arg\min_{\pi} \; \mathbb{E}_{s^*, a^* \sim \pi^*} \; \ell(a^*, \pi(s^*))$$

HOW DO WE IMPLEMENT THIS IN PRACTICE?

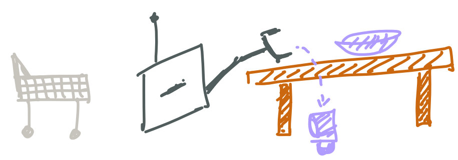
WHAT IS THE LOSS FOR

DISCRETE ACTIONS:

CONTINUOUS ACTIONS:

STEP 3: CHECK VALIDATION LOSS

HOW CAN BC FAIL?



DAGGER

INSIGHT: ASK THE HUMAN EXPERT FOR "CORRECTIONS" ON STATES THE ROBOT VISITS.

$$\arg\min_{\pi} \mathop{E}_{s \sim \pi} \ell\left(\pi^*(s), \pi(s)\right)$$

DAGGER ALGORITHM

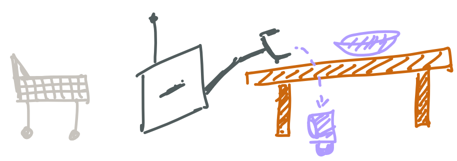
$\pi_0 \leftarrow$ INITIALIZE POLICY WITH BEHAVIOR CLONING.

$D \leftarrow \{\}$ INITALIZE EMPTY DATA BUFFER

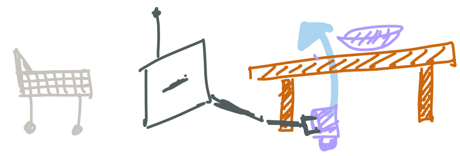
FOR $i = 1 \ldots N$

ROLLOUT π_i

$(s_1, a_1, s_2, a_2, \ldots\ldots)$



QUERY HUMAN π^* FOR CORRECT ACTIONS



$$(s_1, \pi^*(s_1), s_2, \pi^*(s_2), \ldots\ldots)$$

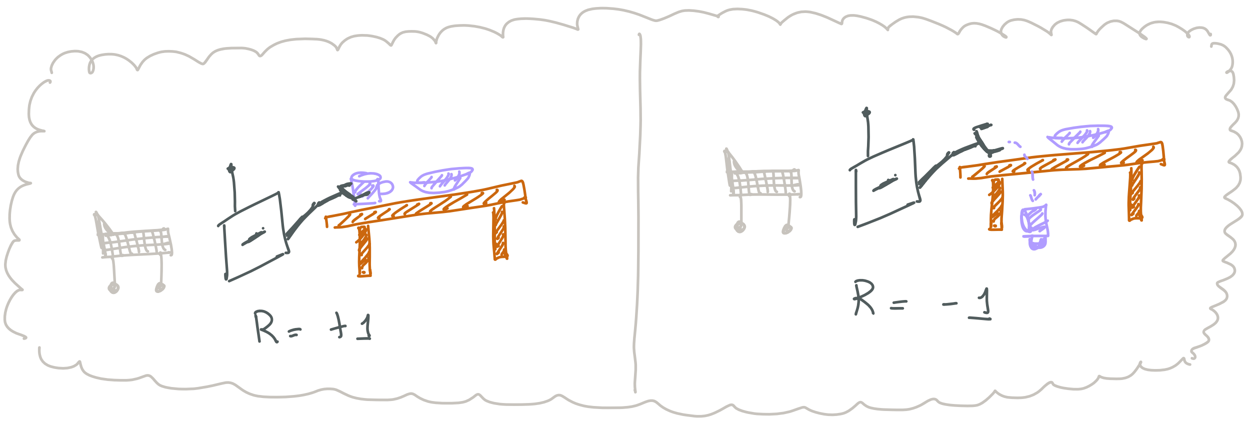
$$D \leftarrow D \cup \{(s_1, \pi^*(s_1), s_2, \pi^*(s_2), \ldots\ldots)\}$$

$$\pi_i \leftarrow \text{TRAIN}(D)$$

PRACTICAL ISSUES WITH DAGGER?

ASK HUMAN TO DESIGN REWARD FUNCTION.

$$\langle S, A, R, T \rangle$$

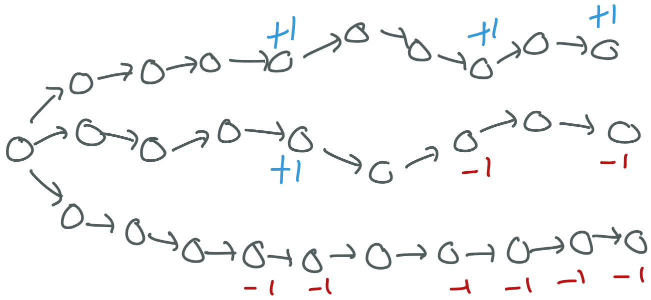


HOW CAN WE TRAIN A POLICY π BY INTERACTING WITH WORLD $T(s'|s,a)$?

POLICY GRADIENT

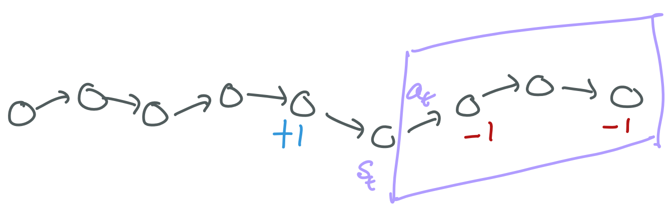
IDEA: ROLLOUT POLICIES, ESTIMATE VALUE, UPDATE.

$$\underset{\pi_\theta}{\arg\max} \; \mathbb{E}_{s_t \sim \pi_\theta} \left[\sum_{t=1}^{T} r(s_t, a_t) \right]$$



POLICY GRADIENT THEOREM

$$\theta' \leftarrow \theta + \eta \left[\sum_{t=1}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t) \; \hat{Q}(s_t, a_t) \right]$$



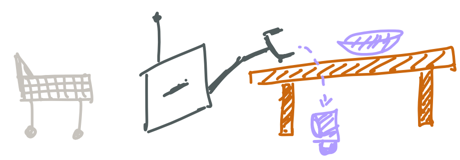
$$\hat{Q}(s_t, a_t) = -1 - 1$$
$$= -2$$

REINFORCE

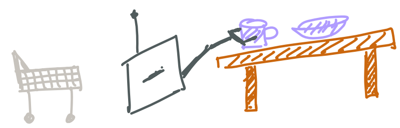
$\pi_0 \leftarrow$ INITIALIZE POLICY

FOR $i = 1 \ldots N$

ROLLOUT π_i:



$(s_1, a_1, r_1, s_2, a_2, r_2, \ldots)$



COMPUTE $\hat{Q}$: (REWARD-TO-GO)

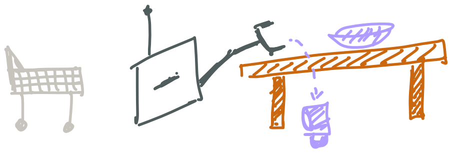
$(s_1, a_1, \hat{Q}_1, s_2, a_2, \hat{Q}_2, \ldots)$

UPDATE POLICY π_θ:

$$\theta' \leftarrow \theta + \eta \left[\sum_{t=1}^{T} \nabla_\theta \log \pi_\theta(a_t | s_t) \, \hat{Q}(s_t, a_t) \right]$$

PROBLEMS WITH REINFORCE ?

WHY IS MODEL-FREE RL CHALLENGING FOR ROBOTICS?



WHAT IF WE LEARN $T(s'|s,a)$?

$$\langle S, A, R, T \rangle$$

WHAT HAPPENS IF WE PLAN WITH A MODEL LEARNED ONLY FROM HUMAN DEMONSTRATION?

$s_t \quad a_t \quad s_{t+1}$

PERFORMANCE DIFFERENCE

$$J_{M^*}(\hat{\pi}) - J_{M^*}(\pi^*)$$

REAL WORLD $\rightarrow$ M^*

$\hat{\pi}$: POLICY THAT PLANNER COMPUTES

π^*: OPTIMAL POLICY

AGNOSTIC SYS IDENTIFIC

$\leq$ MODEL LOSS ON STATES LEARNER VISITS $+$ MODEL LOSS ON STATES EXPERT VISITS $+$ SUB OPTIMALITY OF YOUR PLANNER

DAGGER FOR MODEL BASED RL

GIVEN: DATA FROM AN EXPERT POLICY π^*

$$D_{exp} = \{ (s_i , a_i , s'_i) \}_{i=1}^{N}$$

INIT: INITIALIZE A MODEL $\hat{M}_1(s, a) \rightarrow s'$

$$\mathcal{L}(\hat{M}) = \frac{1}{|D_{exp}|} \sum_{s,a,s' \in D_{exp}} \| \hat{M}(s,a) - s' \|^2$$ Optimize loss

$D_{learner} = \{ \}$ # INITIALIZE LEARNER DATA

FOR $i = 1 \ldots N$ (# OUTER LOOP # N IS ROUNDS OF DATA COLLECT)

CALL A PLANNER ON MY MODEL M_i

$$\hat{\pi}_i = \arg\min_{\pi \in \Pi} J_{M_i}(\pi)$$

COLLECT DATA BY RUNNING POLICY $\hat{\pi}_i$ IN REAL WORLD.

$$D^{i}_{learner} = \{ s_i , a_i , s'_i \}_{i=1}^{N}$$

AGGREGATE LEARNER DATA

$$D_{learner} = D_{learner} \cup D'_{learner}$$

TRAIN MODEL.

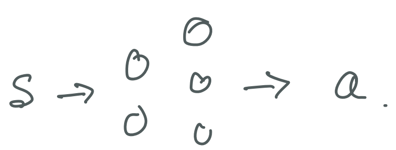
$$\mathcal{L}_1(\hat{M}) = \frac{1}{|D_{learner}|} \sum_{\substack{(s,a,s') \\ \in D_{learner}}} \left\| \hat{M}(s,a) - s' \right\|^2$$

$$\mathcal{L}_2(\hat{M}) = \frac{1}{|D_{exp}|} \sum_{\substack{(s,a,s') \in \\ D_{exp}}} \left\| \hat{M}(s,a) - s' \right\|^2$$

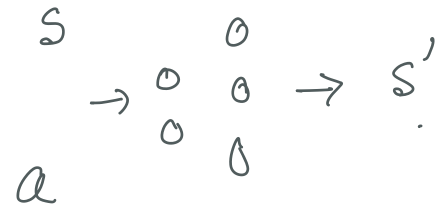
GET NEW MODEL

$$\hat{M}_{i+1} = \text{OPTIMIZE}\left(\mathcal{L}_1(\hat{M}) + \mathcal{L}_2(\hat{M})\right)$$

POLICY



MODEL



D