Train Hard, Fight Easy:
Robust Meta RL

Ido Greenberg, Shie Mannor, Gal Chechik, Eli Meirom

NeurIPS 2023

TECHNION
Israel Institute
of Technology

NVIDIA

Bar-Ilan

Meta Reinforcement Learning

- Learn to adapt to new tasks
- Goal – expected return over all tasks $\{\tau\}$:

$$\mathbf{argmax}\, \boldsymbol{E}_{\boldsymbol{\tau},\boldsymbol{R}}[\boldsymbol{R}]$$

Meta Reinforcement Learning

- Learn to adapt to new tasks
- Goal – expected return over all tasks $\{\tau\}$:

$$\mathbf{argmax}\, \boldsymbol{E}_{\boldsymbol{\tau},\boldsymbol{R}}[\boldsymbol{R}]$$

- Expectation may not suffice
  - Single test task
  - Sensitivity to risk

Robustness in Meta-RL

- Be robust to the task
- Goal – expected return over α lowest-return tasks:

$$\mathbf{argmax}\, \boldsymbol{CVaR_{\tau}^{\alpha}}\left[\boldsymbol{E_R}[\boldsymbol{R}]\right]$$

CVaR Policy Gradient

- Only applied to α worst tasks in the data
  $\Rightarrow$ most data not used
  $\Rightarrow$ **sample inefficient**

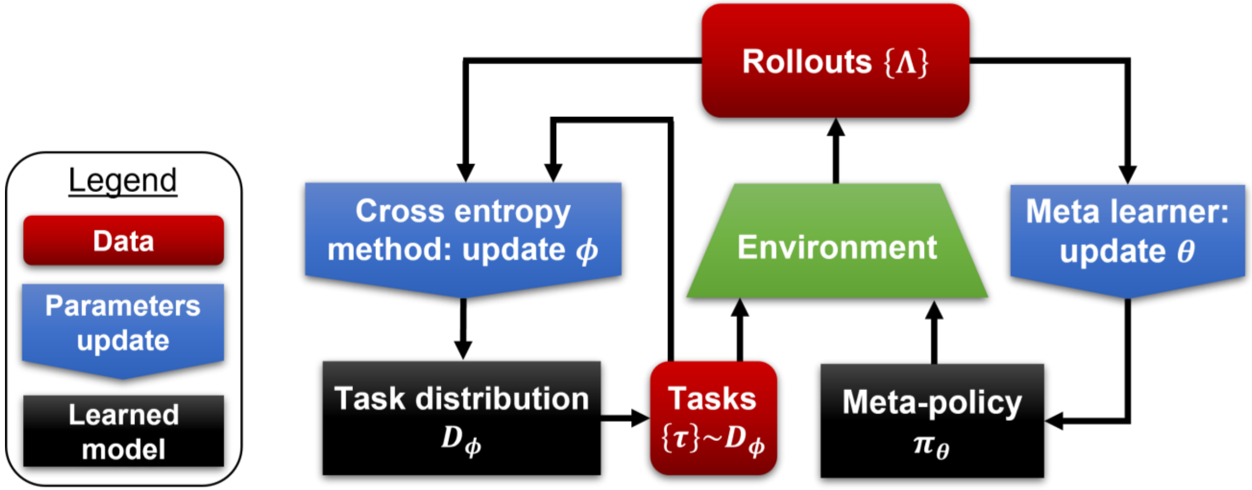
CVaR Policy Gradient

- Only applied to α worst tasks in the data

  $\Rightarrow$ most data not used

  $\Rightarrow$ **sample inefficient**

- Robust Meta-RL (RoML):
  - Over-sample low-return tasks
  - Sample efficiency is recovered



Unbiased Policy Gradient

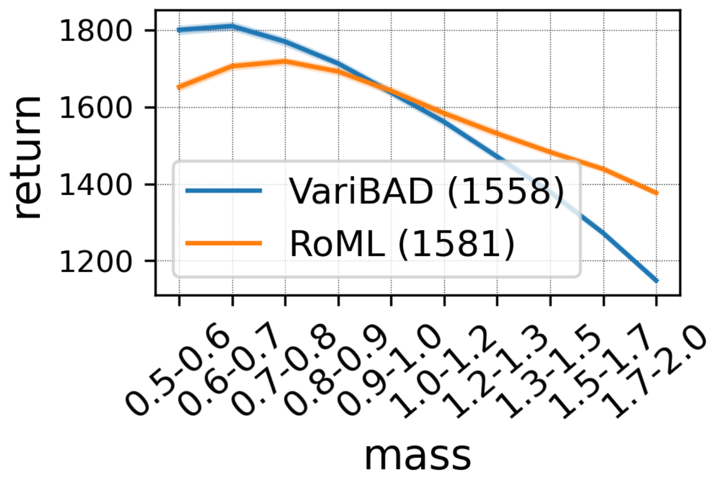
- Background:
  - In standard (non-meta) RL, CVaR-PG yields **biased gradients**

Unbiased Policy Gradient

- Background:
  - In standard (non-meta) RL, CVaR-PG yields **biased gradients**
- Theorem:
  - In **Meta-RL**, CVaR-PG yields **unbiased gradients**

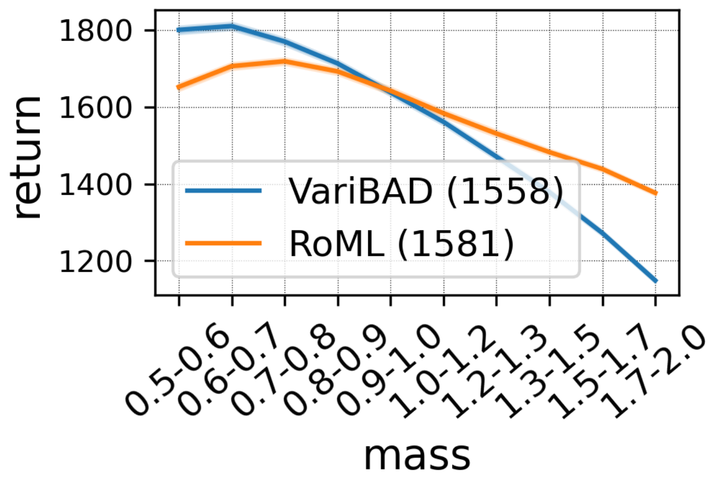
Experiments

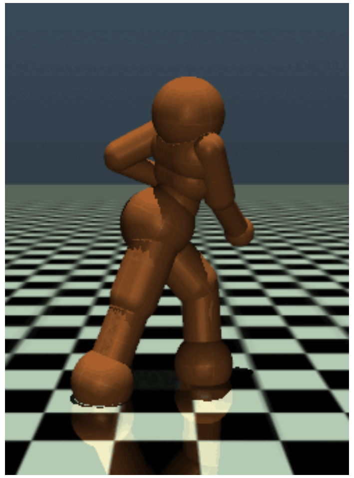
- Better CVaR return
- Less sensitive to task



Experiments

- Better CVaR return
- Less sensitive to task
- Meaningful policies





VariBAD

RoML

RoML as Meta-Algorithm

- Can run on top of any meta-RL algorithm:
  just modify task-sampling in training

- Example: