Taming MAML: Efficient Unbiased Meta-Reinforcement Learning

Hao Liu  Richard Socher  Caiming Xiong
Problematic Gradient Estimation in MAML

- MAML learns a good initialization for stochastic gradient descent adaptation
- **Challenge:** MAML’s gradient involves a sophisticated Hessian which is not easily computable via auto differentiation

\[
\nabla^2_\theta \mathbb{E}_{\tau \sim P_{\tau}|\theta}[R(\tau)] = \\
\int P_{\tau}(\tau|\theta) \nabla^2_\theta \log \pi_\theta(\tau) R(\tau) d\tau + \int P_{\tau}(\tau|\theta) \nabla_\theta \log \pi_\theta(\tau) \nabla_\theta \log \pi_\theta(\tau)^T R(\tau) d\tau
\]

Can be implemented via auto differentiation, e.g. `tf.gradient(tf.gradient(...)...)`  
Missing in existing estimation methods
Computational Efficient Solution: TMAML

Idea: surrogate function + scalable control variates

$$J_{TMAML} = \sum_{t=0}^{H-1} \left( \prod_{t'=0}^{t} \frac{\pi_\theta(a_{t'}|s_{t'})}{\downarrow(\pi_\theta(a_{t'}|s_{t'}))} \right) r(s_t, a_t) + \sum_{t=0}^{H-1} \left[ 1 - \left( \prod_{t'=0}^{t-1} \frac{\pi_\theta(a_{t'}|s_{t'})}{\downarrow(\pi_\theta(a_{t'}|s_{t'}))} \right) \right] \left( 1 - \frac{\pi_\theta(a_t|s_t, z)}{\downarrow(\pi_\theta(a_t|s_t, z))} \right) b(s_t)$$

$$\downarrow$$ denotes ‘stop_gradient’ or ‘detach’

Forward pass: TMAML objective function equals expected reward
Backward pass:

- **Unbiased:** $$\mathbb{E}_{\tau \sim P_{\tau}(\tau|\theta)} [\nabla^2_{\theta} J_{TMAML}] = \nabla_{\theta} \mathbb{E}_{\tau \sim P_{\tau}(\tau|\theta)} [R(\tau)]$$
- **Low variance:** details in paper

Per-task control variates: value function, etc
Meta control variates: learned by MAML itself

Meta control variates is scalable
TMAML reduced meta-gradient variance and improve performance

MAML (Finn et al 2017) is biased

DICE (Foerster et al 2018) is unbiased & high variance  **TMAML is unbiased & low variance**

LVC (Rothfuss et al 2019) is biased & low Variance

Left two figs show meta gradient variance, lower is better, right two figs show corresponding mean reward, higher is better. green and red lines are two versions of TMAML
TMAML outperforms existing methods on most of meta reinforcement learning tasks.

Showing mean reward, higher is better, green and red lines are two versions of TMAML.
Taming MAML: Efficient Unbiased Meta-Reinforcement Learning

Welcome to our poster tonight at Poster #38

Github: https://github.com/lhao499/taming-maml