

Advanced topics in Deep Reinforcement Learning

Week 10 slides

Comparing EvoMAML, REPTILE and ES-MAML in the LunarLander-v2 environment

Thorben Klamt - 30.05.2024

Advanced Topics in Reinforcement Learning, Theresa Eimer, Prof. Dr. rer. nat. Marius Lindauer,
Gottfried Wilhelm Leibniz University Hannover

Comparing EvoMAML, REPTILE and ES-MAML in the LunarLander-v2 environment

EvoMAML

optimizes hyperparameters in meta-reinforcement learning by applying evolutionary strategies to perturb and evolve hyperparameter sets over generations based on performance rewards.

ES-MAML

ES-MAML leverages evolutionary strategies to compute gradients for hyperparameter updates, using perturbations and performance-based rewards to guide optimization in meta-reinforcement learning.

Reptile

Reptile simplifies meta-reinforcement learning by repeatedly sampling tasks, performing gradient updates, and moving the initial parameters towards the updated parameters, effectively learning a good initialization.

Comparing EvoMAML, REPTILE and ES-MAML in the LunarLander-v2 environment

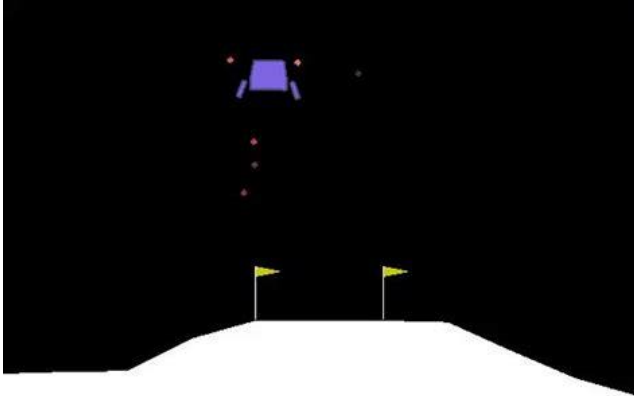


Fig.1: Stable baselines lunar lander-v2

EvoMAML preliminary result

-175.63824800000003
-120.4455406
-135.71329419999998
-110.52858959999999
-80.7996916

ES-MAML preliminary result

-175.63824800000003
3.2476257000000004
-59.981305600000006
24.608173400000005
44.400115099999994

Reptile preliminary result

-882.2719797
-826.1661773
-899.4343997
-1728.1372563
-494.35589730000004

```

import tensorflow as tf
import torch

# Check GPU availability
print("Num GPUs Available: ", len(tf.config.list_physical_devices('GPU')))

def set_seed(seed):
    np.random.seed(seed)
    tf.random.set_seed(seed)
    torch.manual_seed(seed)

set_seed(1)

# Enable mixed precision training if supported
from tensorflow.keras.mixed_precision import set_global_policy

policy = tf.keras.mixed_precision.Policy('mixed_float16')
set_global_policy(policy)

# Define the environment
env_name = "LunarLander-v2"

# Create a vectorized environment
num_envs = 16 # Number of environments to run in parallel
env = make_vec_env(env_name, n_envs=num_envs, vec_env_cls=DummyVecEnv)
    
```

Fig.2: Screenshot of the implementation