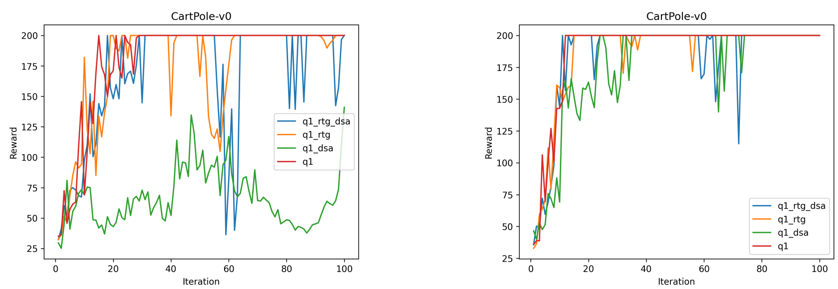
Assignment 2: Policy Gradients

Yulun Rayn Wu, 3034358565

October 20, 2020

1 Small-Scale Experiments

- Experiment 1



(a) batch size: 1000 **(b)** batch size: 5000

Figure 1: Batch: 1000 vs. 5000. **Settings**: env_name: CartPole-v0; n_iter: 200; eval_batch_size: 400; num_agent_train_steps_per_iter: 1; discount: 1.0; learning_rate: 5e-3; n_layers: 2; size(hidden layer): 64; seed: 1.

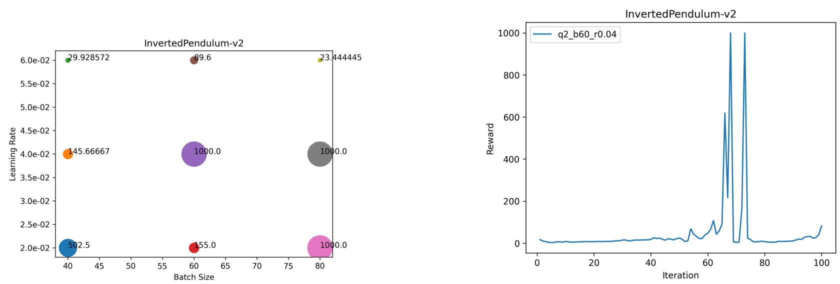
  - Without advantage standardization, using reward-to-go yields significantly better performance than the trajectory-centric one.
  - Advantage standardization helped in both the trajectory-centric and reward-to-go case, especially the former.
  - Experiments with larger batch size converges faster in terms of iteration and achieve more stable results.

Configurations:

```
python cs285/scripts/run_hw2.py --env_name CartPole-v0 -n 100 -b 1000 \
    -dsa --exp_name q1_sb_dsa
python cs285/scripts/run_hw2.py --env_name CartPole-v0 -n 100 -b 1000 \
    -rtg -dsa --exp_name q1_sb_rtg_dsa
python cs285/scripts/run_hw2.py --env_name CartPole-v0 -n 100 -b 1000 \
    -rtg --exp_name q1_sb_rtg
python cs285/scripts/run_hw2.py --env_name CartPole-v0 -n 100 -b 1000 \
    --exp_name q1_sb
```

```
python cs285/scripts/run_hw2.py --env_name CartPole-v0 -n 100 -b 5000 \
    -dsa --exp_name q1_lb_dsa
python cs285/scripts/run_hw2.py --env_name CartPole-v0 -n 100 -b 5000 \
    -rtg -dsa --exp_name q1_lb_rtg_dsa
python cs285/scripts/run_hw2.py --env_name CartPole-v0 -n 100 -b 5000 \
    -rtg --exp_name q1_lb_rtg
python cs285/scripts/run_hw2.py --env_name CartPole-v0 -n 100 -b 5000 \
    --exp_name q1_lb
```

- Experiment 2



(a) hyperparameter search **(b)** -b 60 -lr 4e-2

Figure 2: Search: batch_size, learning_rate. **Settings**: env_name: InvertedPendulum-v2; n_iter: 100; eval_batch_size: 400; num_agent_train_steps_per_iter: 1; ep_len: 1000; discount: 0.9; n_layers: 2; size(hidden layer): 64; seed: 1.

  - The batch size and learning rate found are 60 and 4e-2.

Configurations:

```
python cs285/scripts/run_hw2.py --env_name InvertedPendulum-v2 \
    --ep_len 1000 --discount 0.9 -n 100 -l 2 -s 64 -b 60 -lr 4e-2 -rtg \
    --exp_name q2_b60_r0.04
```

2 More Complex Experiments

- Experiment 3

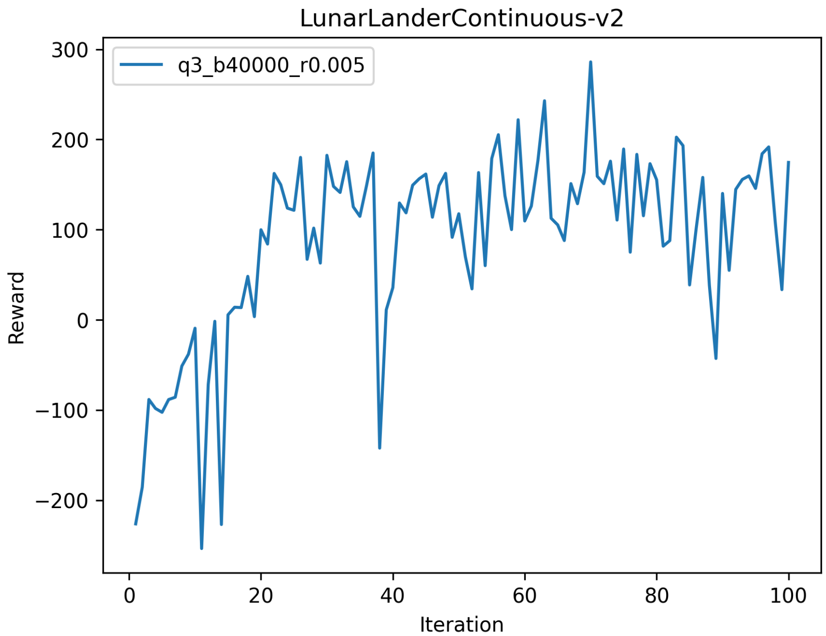
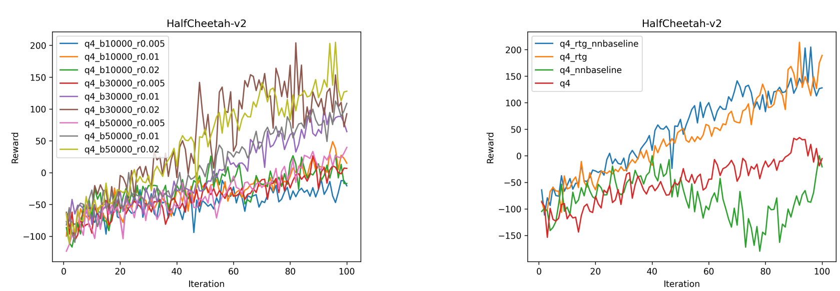


Figure 3: Settings: env_name: LunarLanderContinuous-v2; n_iter: 100; eval_batch_size: 400; num_agent_train_steps_per_iter: 1; ep_len: 1000; discount: 0.99; batch_size: 40000; learning_rate: 5e-3; n_layers: 2; size(hidden layer): 64; seed: 1.

- Experiment 4



(a) hyperparameter search **(b)** -b 50000 -lr 2e-2

Figure 4: Search: batch_size, learning_rate. **Settings**: env_name: HalfCheetah-v2; n_iter: 100; eval_batch_size: 400; num_agent_train_steps_per_iter: 1; ep_len: 150; discount: 0.95; n_layers: 2; size(hidden layer): 32; seed: 1.

  - The batch size and learning rate found are 50000 and 2e-2.

3 Bonus

- GAE-λ for advantages estimation

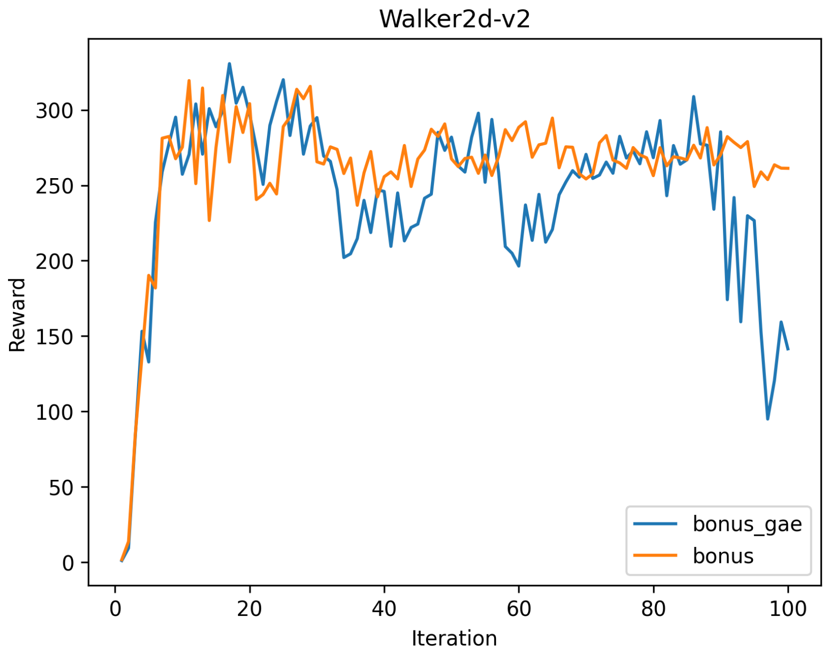


Figure 5: Settings: env_name: Walker2d-v2; n_iter: 100; eval_batch_size: 5000; num_agent_train_steps_per_iter: 1; ep_len: 1000; discount: 0.99; batch_size: 10000; learning_rate: 5e-3; n_layers: 2; size(hidden layer): 64; seed: 1.

- There is no evidence that GAE-λ would speed up the training process from the result of this experiment.