Baselined REINFORCE

Exercise 7.2: Baselined REINFORCE

In this exercise, you will implement the Baselined REINFORCE algorithm.

Make sure that you have completed the setup requirements as described in the Set Up Lab Environments section.

Now, run jupyter notebook and open the "Ex7.2 Baselined REINFORCE.ipynb" notebook.

Examine the notebook. We have given you boiler plate and helper code for the implementation of the Baselined REINFORCE algorithm. Basically, the basic REINFORCE algorithm is implemented and you need to define a critic network, and then modify the associated trainer and loss function accordingly.

Once you got yourself acquainted with the notebook, go to #TODO 1. Here you need to define a critic network that learns the value function V(s). The CNTK syntax is given. You need to determine the input and output of this critic network.

Lab Question

1/1 point (graded)

What should be assigned to the critic_input?

- 0
- 1

- 2
- 4
- 128 ✔

Submit You have used 1 of 2 attempts

✔ Correct (1/1 point)

Lab Question

1/1 point (graded)

What should be assigned to the critic_output?

- 0
- 1 ✔
- 2
- 4
- 128

Submit You have used 1 of 2 attempts

✔ Correct (1/1 point)

Next go to #TODO2. Here you need to define the trainer for the critic. To do that, you need to define the critic loss function and the critic target.

Lab Question

1/1 point (graded)

What could be an example of a critic target for this context?

- (●) critic_target = C.sequence.input_variable(1, np.float32, name="target") ✔
- ( ) critic_target = C.sequence.input_variable(state_dim, np.float32, name="target")
- ( ) critic_target = C.sequence.input_variable(hidden_size, np.float32, name="target")
- ( ) critic_target = C.sequence.input_variable(action_count, np.float32, name="target")

Submit You have used 1 of 2 attempts

✔ Correct (1/1 point)

Lab Question

1/1 point (graded)

What could be an example of a critic loss for this context?

- ( ) critic_loss = C.times(critic, critic_target)
- ( ) critic_loss = C.mean(critic, critic_target)
- ( ) critic_loss = C.log(critic, critic_target)
- (●) critic_loss = C.squared_error(critic, critic_target) ✔

Submit You have used 1 of 2 attempts

✔ Correct (1/1 point)

Now, go to #TODO3. Here you need to train the critic to predict the discounted reward from the observation.

Lab Question

1/1 point (graded)

Which code example can you use to train the critic network for this context?

- ○ critic_trainer.train_minibatch({observations: epl, critic_target: discounted_epr})
- ○ critic_trainer.train_minibatch({observations: epr, critic_target: discounted_epr})
- ◉ critic_trainer.train_minibatch({observations: epx, critic_target: discounted_epr}) ✔
- ○ critic_trainer.train_minibatch({observations: baseline, critic_target: discounted_epr})

Submit You have used 1 of 2 attempts

✔ Correct (1/1 point)

Lastly, predict the discounted reward using the eval() function of the critic network and assign it to baseline.

Lab Question

1/1 point (graded)

Which code example can you use to perform the above task for this context?

- ○ baseline = critic.eval({observations: epl})
- ○ baseline = critic.eval({observations: epr})
- ○ baseline = critic.eval({observations: discounted_epr})

baseline = critic.eval({observations: epx}) ✔

Submit You have used 1 of 2 attempts

✔ Correct (1/1 point)

You now have an implementation of the Baselined REINFORCE algorithm. Run this notebook several times and use max_number_of_episodes = 500.

Lab Question

1/1 point (graded)
Based on your observation of the above experiments, on average, does the agent manage to reach the goal within 500 episodes?

No

Yes ✔

Submit You have used 1 of 2 attempts

✔ Correct (1/1 point)

Lab Question

1/1 point (graded)
Based on your observation of the above experiments, on average, what is the variance when the agent reach the goal?

Between 0 to 500

Between 500 to 1000

- (●) Between 1000 to 2000 ✔
- ( ) Above 2000

Submit You have used 1 of 2 attempts

✔ Correct (1/1 point)