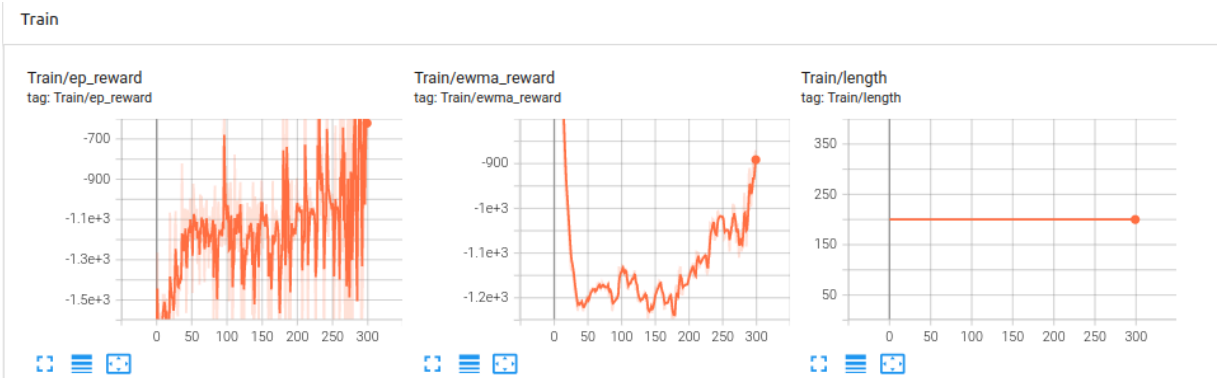
RL Problem 4

experiment (a)

Tensorboard

ep_reward & emma_reward & length



hyperparameters

Layers

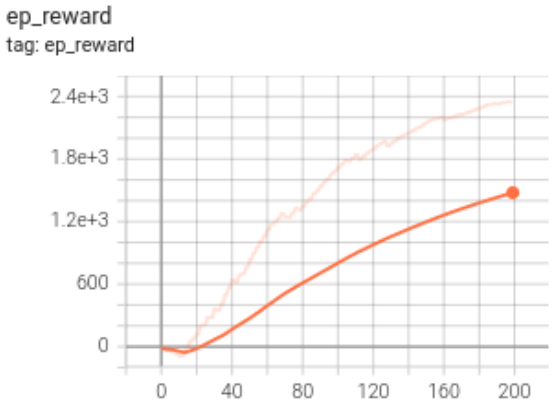
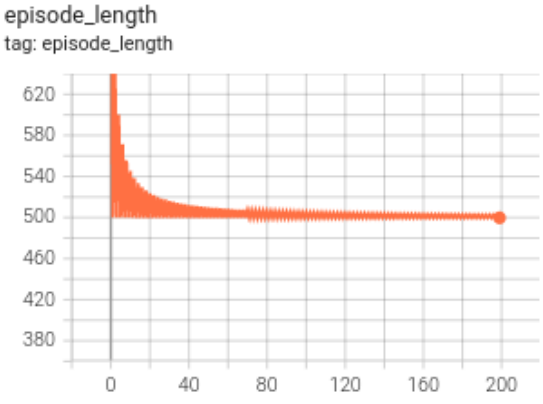
```
self.actor = nn.Sequential(
    nn.Linear(num_inputs, hidden_size),
    nn.ReLU(),
    nn.Linear(hidden_size, hidden_size),
    nn.ReLU(),
    nn.Linear(hidden_size, num_outputs),
    nn.Tanh()  # Using Tanh activation fo
)
# Construct your own critic network
self.critic = nn.Sequential(
    nn.Linear(num_inputs + num_outputs, hidden_size),
    nn.ReLU(),
    nn.Linear(hidden_size, hidden_size),
    nn.ReLU(),
    nn.Linear(hidden_size, 1)
)
```

hyperparameters

```
num_episodes = 300
gamma = 0.997
tau = 0.006
hidden_size = 128
noise_scale = 0.40
replay_size = 100000
batch_size = 128
updates_per_step = 1
print_freq = 1
ewma_reward = 0
rewards = []
ewma_reward_history = []
total_numsteps = 0
updates = 0
max_action = 2  # for Pendulum
```

experiment(b)

Tensorboard

ep_reward	length
	

hyperparameters

actor, critic

```
self.layers = nn.Sequential(
    nn.Linear(num_inputs, hidden_size),
    nn.ReLU(),
    nn.Linear(hidden_size, hidden_size),
    nn.ReLU(),
    nn.Linear(hidden_size, num_outputs),
    nn.Tanh()
)
```

```
self.layers = nn.Sequential(
    nn.Linear(num_inputs + num_outputs, hidden_size),
    nn.ReLU(),
    nn.Linear(hidden_size, hidden_size),
    nn.ReLU(),
    nn.Linear(hidden_size, 1)
)
```

hyperparameters

```
num_episodes = 200
gamma = 0.995
tau = 0.002
hidden_size = 128
noise_scale = 0.1
replay_size = 100000
batch_size = 128
updates_per_step = 1
print_freq = 2
ewma_reward = 0
rewards = []
ewma_reward_history = []
total_numsteps = 0
updates = 0
```