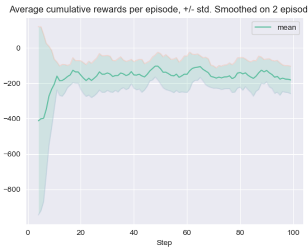
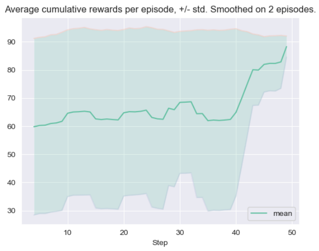
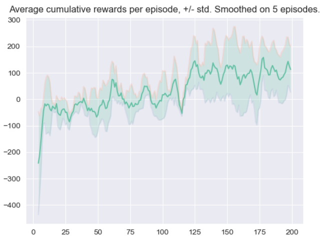
TME #07: Continuous Actions

In this TME, we implement a DDPG agent for environments with continous actions, and test its performance on 3 environments with continuous action spaces: MountainCarContinuous-v0, LunarLanderContinuous-v2 and Pendulum-v0. We look at the evolution of the reward

In DDPG, the actor network is a deterministic mapping from states to actions. The exploration of the environment is assured by sampling actions from a normal distribution centered on the output of the actor network. The standard deviation *Sigma* can be maintained as a diagonal matrix with entries decreasing towards zero as the model learns. We find that for Mountaincar, maintaning a high level of exploration as long as possible is crucial, and keep a constant exploration rate on that environment. For the 2 others, the standard deviation of the normal distribution decreases at each time step with a rate of 0.99.

We use 2 simple fully connected neural networks with 2 hidden layers for the critic and the actor. Provided that enouch transitions have been stored in memory, the networks are updated at each time step. Like for DQN, the targets of the critic are computed with target networks, which are updated every 100 episodes by polyak averaging with a parameter of 0.99.

We report the average of the cumulative rewards (including both train and test mode) evolution for each time step over 5 different experiments launched with different seeds. Each experiment lasts 50, 100 and 200 episodes for Mountaincar, Pendulum and Lunar respectively.

Pendulum-v0	MountainCarContinuous-v0	LunarLanderContinuous-v2
		

On average, the `MountainCar` experiments needs a lot of exploration before the learning peak around 80 episodes. The average being positive (around a cumulative reward of 60) before indicates that more often than not, the final reward of 100 is found at the first episode. But only after 40 episodes does it happen *consistently*. This relatively high variance is reflected in the size of he uncertainty interval.

On the other hand, on the `Pendulum` environment, it takes around 10 episodes for the mean accross all seeds to oscillate around -150, which we consider solved. The agent then stays remarkably stable for the next 100 episodes in all 5 experiments.

The `Lunar` experiment is the most unstable. Our DDPG agent consistently reaches a zone of cumulative rewards of 200, but encounters frequent collapses. We suspect that this environment is highly sensible to the local choice of a suboptimal action (under the exploration policy).