Least Squares Methods for DRL – Project Game Plan

Goal: get into the field of Reinforcement Learning, learn advanced coding in PyTorch, implement algorithms from papers, suggest improvements to algorithms, learn to connect between "shallow" and "deep" methods, gain deep understanding of DQN and DDPG.

Tools: Python, PyTorch, Simulators: OpenAI's Atari, BiPedal-Walker, PyBullet (3D robots)

Recommended Steps & Resources:

1. Introduce yourself to RL with these courses:
    a. RL Course by David Silver (DeepMind) - https://www.youtube.com/watch?v=2pWv7GOvuf0&list=PL7-jPKtc4r78-wCZcQn5IqyuWhBZ8fOxT
    b. Stanford's CS234- Reinforcement Learning - https://www.youtube.com/watch?v=FgzM3zpZ55o&list=PLoROMvodv4rOSOPzutgyCTapiGlY2Nd8u
2. Learn the basics of PyTorch. It has great tutorials, also in RL, and pretty simple. Checkout https://pytorch.org/tutorials/. Work with the latest versions (PyTorch 1.1).
3. Papers and articles to read:
    a. DQN - https://deepmind.com/research/dqn/
    b. Double DQN - https://arxiv.org/abs/1509.06461
    c. Dueling DQN - https://arxiv.org/abs/1511.06581
    d. DDPG - https://arxiv.org/abs/1509.02971
    e. Shallow Updates for DRL (the original work) - https://arxiv.org/abs/1705.07461
    f. Least Squares Methods for DRL Project (Tal) - https://github.com/taldatech/pytorch-ls-dqn/blob/master/writeup/Deep_RL_Shallow_Updates_for_Deep_Reinforcement_Learning.pdf
    g. D4PG - https://arxiv.org/abs/1804.08617
4. Implement the algorithm for D4PG, **or any other DRL algorithm that uses a Replay Buffer**, run test on environments of your choosing (OpenAI/PyBullet). You can use my previous implementations in PyTorch. Please write a clean code (variable names are understandable, every function has a description. Take a look at my code for example).
    a. LS-DQN (with DoubleDQN, DuelingDQN, boosting) - https://github.com/taldatech/pytorch-ls-dqn
    b. LS-DDPG - https://github.com/taldatech/pytorch-ls-ddpg
5. Document every result you get, use graphs/tables/drawings…
6. Upload everything to a well-documented (with README.md) GitHub repository. You can add me as collaborator (taldatech) if you need any help from me or you want me to take a look at your code.