ECEN 689: RL: Fall 2019
Assignment 3

In this homework you will implement Value Iteration, Policy Iteration and Q-Learning in the frozen lake environment from openAI Gym (Link).

Setup Instructions

1. Ensure that you are using Python 3.5+ (for Gym) and have the following libraries/dependencies: time, seaborn, matplotlib.pyplot, numpy, math, random
   (a) People working with Windows can find solace using Anaconda (Link).
   (b) People working with Mac OS or any Linux OS have an inbuilt python at their disposal. So, just follow the installation procedure given in openAI website.
2. Refer this for installing gym environment.
3. This homework comes with a helper file (A3Helper.ipynb), it consists of a few base codes that will be useful.

Questions

1. **Value Iteration**
   - Implement value iteration in the frozen lake environment. You will have to make use of the transition kernel.
   - Plot $||V_k - V_{k-1}||_2$, where k is the iteration number. This will show the convergence of your algorithm.
   - Compute the optimal value function and the optimal policy.
   - Use the *fancy_visual* function, which is available in the helper file to plot the optimal value function and policy.
   - Compute and print the optimal Q function.
2. **Policy Iteration**
   - Implement policy iteration.
   - Plot $||V_{\pi_k} - V_{\pi_{k-1}}||_2$
   - Plot the optimal value function and policy using the *fancy_visual* function.
   - Print the optimal Q function.

3. **Tabular Q Learning**

   - Implement tabular Q learning (no function approximation) on the frozen lake environment. Here you *shall not* use the transition kernel. The helper file has some basic examples on sampling the environment.
   - Plot $R_{total,k}$, where $R_{total,k}$ is the cumulative reward obtained in episode k.
   - Plot $||Q_k - Q^*||_2$ where k is the episode number and Q^* is the optimal Q function obtained from 1 or 2.
   - Redo the above two plots, but now with a sliding average window (generalize window length for testing).
   - Compute the final policy and plot the heat map.
   - *Hint: Play with different learning rates and exploration rates. This will help you converge in lesser number of episodes.*

Submission

1. Create three separate IPython Notebooks (.ipynb) files. They should contain all your plots and outputs. Name the files as follows,

   - VI.ipynb for value iteration (should include of 2 plots and the optimal Q-function, value function, and policy).
   - PI.ipynb for policy iteration (should consist of 2 plots and the optimal Q-function, value function, and policy).
   - QL.ipynb for Q learning (should consist of 5 plots and the final Q-function, value Function, and policy).

2. While it is a good practice to print out intermediate steps to debug your code, please refrain from doing so in the final submission.

3. You may choose to make your code modular, in that case please submit all relevant files.

4. Your submission should consist of the following files,

   - All the files in 1.
   - HTML versions of all the files in 1 (similar to Homework 1).
   - All relevant files if 3 applies to you.

5. Zip all files in 4 and name it **LastName_FirstName.zip** and submit it via eCampus.