CIS 620, Advanced Topics in Deep Learning, Spring 2024

Homework 6
Due: Monday, March 11, 11:59pm
Submit to Gradescope

Task 1: Project Proposals!

Submit your team information and proposal through ([Link](#)).

Each team only needs to submit **ONCE**.

Task 2: BCO from Expert

Experiment with and evaluate BCO model trained on an expert.

Learning Objectives

In this assignment, you will:

- Train a PPO agent and use it as an expert for BCO
- Experiment with changes in environment
- Learn about fine-tuning RL models

Deliverables

This is a **pair** assignment for both the written and coding portions. - **One submission per pair**

1. **A PDF with your name in the agreement**

   Copy and edit this googledoc to enter your names in the Student Agreement and answer the questions mentioned below. **Don't forget to add your team member on gradescope submission!**

2. **hw6.ipynb file with the functional code**

   Complete the coding assignment in the Jupyter Notebook and upload the file to Gradescope. **Don't forget to add your team member on the gradescope submission!**

Note that there is a separate assignment for the papers, which will be handed in separately. Due after spring break.

Homework Submission Instructions

Written Homeworks

All written homework must be submitted as a PDF to Gradescope. **Handwritten assignments (scanned or otherwise) will not be accepted.**

Coding Homeworks

All coding assignments will be done in Jupyter Notebooks. We will provide a .ipynb template for each assignment as well as function stubs for you to implement. You are free to use your own installation of Jupyter for the assignments, or Google Colab, which provides a Jupyter Environment connected to Google Drive along with a hosted runtime containing a CPU and GPU.

For this assignment, we have provided the code for dataset creation from a single image. The input image is provided in the folder and can also be accessed by the code for dataset creation.

Questions

1. Train a PPO agent using Stable Baselines to Perform Behavioral Cloning.
   a. Explain all the parameters that are shown during the training of the PPO agent. I used a three-layer (hidden size: 10) MLP model to clone the behavior of the expert model and used an Adam optimizer with a learning rate = 0.01.

      I trained the model for 10 epochs with batch_size = 128.

   b. How are you creating the dataset for Behavioral Cloning?
      I randomized the configuration of the environment and recorded the action the expert took. In the code, I ran the randomization and the collection steps 10 times resulting in a dataset with a size of 5000.

   c. How is the performance of the model compared to the expert?
      The model performs super well in the original environment; the model performs similarly to the expert.

2. Change the parameters of the CartPole environment and test your model.
   a. How does the trained BCO model perform?
      The BCO model has a good performance in the new environment, but it cannot control the position of the cart to let the cart stay in the middle of the environment.

   b. Do you think the PPO model will perform better in this new environment?
      Yes, I think the PPO model will better control the position.
      If the BCO model makes an error in the new environment, the BCO model might act incorrectly in the state it has never seen. (But it will be as stable as PPO for most of the time.

3. Describe two strategies specifying how you can fine-tune your BCO model for this new environment. Mention the specifics i.e algorithm, loss etc.
   - We can let the expert generate the state-action pairs in the new environment and do Behaviour cloning based on the current BCO model.
   - We can try to construct a loss function:
         Loss:(angle_t+1 - angle_t)**2 + (position_t+1 - position_t)**2
     Where angle_t+1 is the angle of the pole after act time t with observing state S_t.
     The position is how far the cart deviates from the center of the environment.
     This allows us to train directly on the current BCO model.

Optional (*not required*) but fun!
   4. Implement one of your strategies and test your results!