Multi-Task Offline Reinforcement Learning with Conservative Q-Learning

Author: Laimonas Lipinskas
Contact: l.lipinskas@student.tudelft.nl

Responsible professor: Matthijs Spaan
Supervisor: Max Weltevrede

1. Background

- **Reinforcement Learning (RL)**: A type of machine learning where the agent learns by interacting with an environment and receiving rewards.
- **Offline RL**: A type of **RL** where the agent learns from a dataset of interactions collected from a live environment, with the goal of creating a policy that matches or outperforms the one used for data collection. This approach is useful when live interaction is too costly.
- **Multi-task RL**: when an **RL** model is trained on multiple tasks and is deployed on either familiar or entirely new tasks.
- **Behavior Cloning** (**BC**): A machine learning method that learns by imitating observed interactions.
- **Q-Learning**: An RL algorithm that learns the value of actions in a given state to maximize total reward.
- **Conservative Q-Learning** (**CQL**): A variant of **Q-Learning** that prioritizes safety and reduces the risk of overestimation of action values.

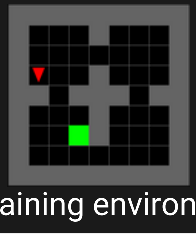
2. Motivation & Research Question

A recent study by Mediratta et al. [1] has shown that modern **offline RL** methods do not outperform **BC** in a **multi-task** setting when it comes to generalizing to different tasks. This raises the question: are these advanced **offline RL** algorithms worth using if they cannot surpass simple imitation?

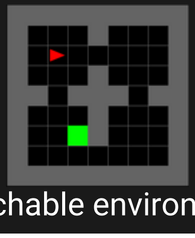
This study aims to extend the experiment conducted by Mediratta et al. [1] to a different environment by specifically comparing **BC** and **CQL**, while also examining the effects of more diverse and larger dataset sizes.

3. Methodology & Experimental Setup

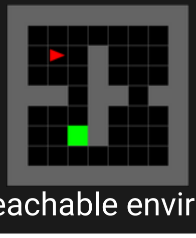
- The experiment is held in an **RL** environment provided by Minigrid [3].
- Training and testing environments can have a different layout of doors, agent starting.
- The models, provided by d3rlpy [4], are trained on datasets of sizes **1000**, 5000,, **10000** 25000, and **100000** collected by policies:
  - **Expert**: always takes the optimal action towards the goal. This is the baseline.
  - **Expert-Suboptimal:** has a 50% chance of taking a suboptimal action and 50% to take an optimal action.
  - **Random Walk:** takes 10 to 50 random actions and after follows a path of optimal actions. A single optimal path is also added for each environment.
- To test the ability of the agent to generalize they are evaluated on:
  - **Reachable tasks [2]**: environments that have different starting positions for the agents when compared to the training set.
  - **Unreachable tasks [2]:** environments that have the same starting and goal locations but with a changed placement of doors.



A training environment



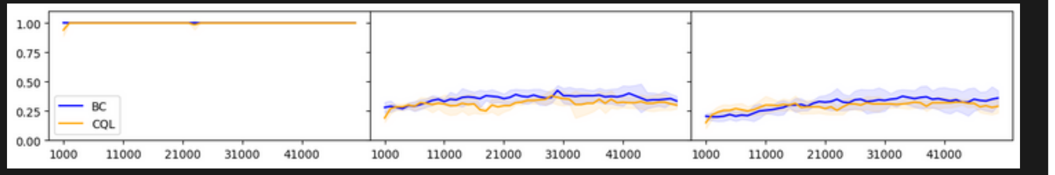
A reachable environment



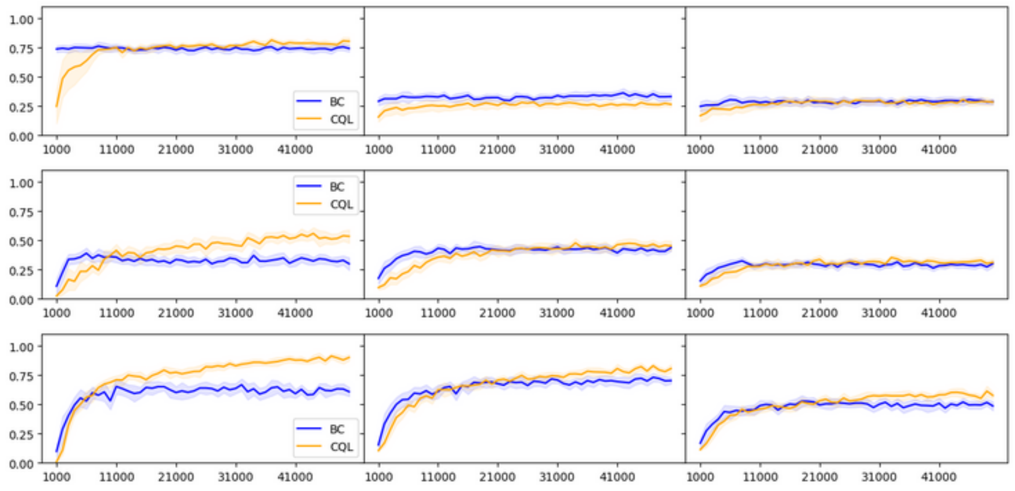
An unreachable environment

4. Results

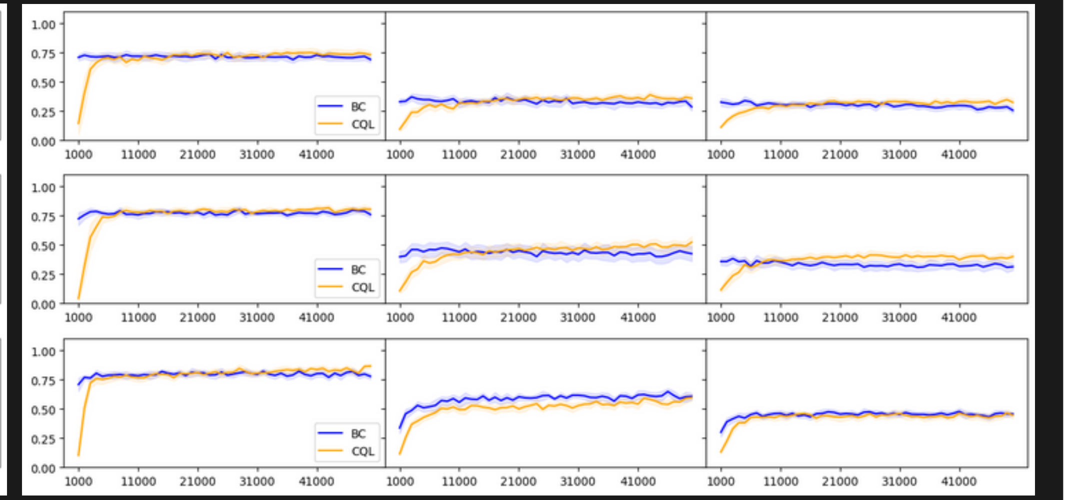
The graphs shown in rows are average rewards from the **training** (**Left column**), **reachable** (**Middle column**) and **unreachable** (**Right column**) environment sets. The maximal attainable reward is 1, meaning that the agent reached the goal in every environment. **Orange** indicates **CQL**, **Blue** indicates **BC**.



Results for models trained on the **Expert** policy on a dataset of size **372**



Results for models trained on the **Random Walk** policy on datasets of size **1000** (**Upper row**), **10000** (**Middle row**) and **100000** (**Lower row**)



Results for models trained on the **Expert-Suboptimal** policy on datasets of size **1000** (**Upper row**), **10000** (**Middle row**) and **100000** (**Lower row**)

5. Conclusions & observations

- In terms of generalization, **BC** did not outperform **CQL**, as the results for both methods were largely equal on both **reachable** and **unreachable** environments.
- Data diversity helped the methods generalize better but only at larger dataset sizes.
- **CQL** had better **training** performance on the **Random Walk** datasets. Though this lead to a larger generalization gap than **BC**'s, it also had the highest mean reward seen in the study for the test sets.

6. Future Work

The generalization gaps of the methods could be compared more precisely in a similar experiment to the one used in Weltevrede et al. [2]. The experiment could use data collection policies that start off by gathering random data and then shift over time to more optimal data. By tuning this shift, it could possibly reveal how data diversity correlates with performance in generalization.

References

[1] Ishita Mediratta, Qingfei You, Minqi Jiang, and Roberta Raileanu. The generalization gap in offline reinforcement learning, 2024

[2] Max Weltevrede, Matthijs T. J. Spaan, and Wendelin BÃ¶hmer. The role of diverse replay for generalisation in reinforcement learning, 2023

[3] Takuma Seno and Michita Imai. d3rlpy: An offline deep reinforcement learning library. Journal of Machine Learning Research, 23(315):1–20, 2022

[4] Maxime Chevalier-Boisvert, Bolun Dai, Mark Towers, Rodrigo de Lazcano, Lucas Willems, Salem Lahlou, Suman Pal, Pablo Samuel Castro, and Jordan Terry. Min igrid & miniworld: Modular & customizable reinforcement learning environments for goal-oriented tasks. CoRR, abs/2306.13831, 2023