**Abstract**

- Existing offline Reinforcement Learning (RL) algorithms aim to leverage previously collected datasets to learn effective policies without further exploration. However, in practice, the distributional shift between the learned policy and the policy used to collect the data often leads to overestimation.
- A recently proposed offline RL algorithm (CQL), showed promising results by adding a regularization term to prevent overestimation.
- In our study we experimented with different regularization methods to improve on the implementation of the CQL algorithm.

**Methods and Materials**

We applied regularization to the CQL algorithm’s loss function to both achieve pessimism and avoid over-fitting.

\[
\ell(\theta_t) = \alpha \sum_{s,a} \log \sum_a \exp Q_{\theta_t}(s, a) - \sum_{s,a} \log Q_{\theta_t}(s, a) - \gamma \ell_{\text{reg}}(\theta_t)
\]

- First Method: Using the square of the Q function in the logarithmic exponential function.
- Second Method: Using the square root of the Q function in the logarithmic exponential function.
- Third Method: Changing the loss function to a mean square function.

**Results**

- We evaluated the algorithm on a variety of robotic tasks from the D4RL library.
- Methods 1 and 3 marginally outperformed the original implementation of CQL on the TD Error evaluation for the Hopper task.
- Method 2 outperformed CQL by a significant margin on the TD error evaluation and slightly underperformed on the environment metric.

**Discussion**

- The most significant results were achieved in continuous domain tasks, particularly on the TD error metric.
- Although the results of our experiment are promising, we only able to evaluate our algorithm in a few D4RL environments and for a limited number of training iterations.

**Conclusions and Future Directions**

- The video rendering of our experiments suggest some improvement in performance in some cases but further testing and analysis is needed for a definitive conclusion.
- Testing our regularization methods on more tasks, including the D4RL library and other robotic task simulators.
- Experimenting with other regularization methods.
- Investigating the reason behind our methods poor performance in discrete domain tasks.

**Acknowledgements**

- My mentor Tianhao Wu, for guiding me through the project and choosing this topic that I really enjoyed.
- Our PI Jiantao Jiao, for providing us with all the resources needed for the project.
- Transfer-to-Excellence program, for giving me the opportunity and the resources to work on this project.
- The Hopper-Dean Foundation, for funding this project.

**References**