SCDAA Coursework Report 2024-25

Introduction

This report details our implementation of an actor-critic algorithm for solving linear quadratic regulator (LQR) and soft LQR problems. The coursework was broken into exercises that build up from solving the Riccati ODE to implementing a full actor-critic method.

Exercise 1: Strict LQR Problem

- Objective: Solve the LQR problem via the Riccati ODE.
- Implementation: The LQRProblem class numerically integrates the Riccati equation and provides methods for evaluating the value function and optimal control.
- Verification: Monte Carlo simulations (in monte_carlo_simulation.py) confirm the expected convergence rates in both time discretization and sample size.

Exercise 2: Soft LQR Problem

- Objective: Extend the LQR formulation by including an entropy regularization term.
- Implementation: The SoftLQRProblem class adapts the Riccati ODE using an effective control cost and provides a stochastic (Gaussian) optimal control policy.
- Analysis: Trajectory comparisons show that as the entropy regularization parameter decreases, the soft LQR converges to the strict LQR solution.

Exercise 3: Critic-Only Algorithm

- Objective: Learn the value function for the soft LQR problem using simulated data.
- Implementation: A neural network (OnlyLinearValueNN) is trained to minimize the temporal difference error.
- Results: The critic loss decreases over training epochs, achieving reasonable accuracy compared to the analytic solution.

Exercise 4: Actor-Only Algorithm

- Objective: Learn the optimal policy using the known value function as a baseline.
- Implementation: The PolicyNet network is trained via policy gradient updates. The network outputs a linear feedback matrix that defines the mean of the Gaussian policy.
- Results: Training drives the policy to produce actions that closely match the optimal ones derived analytically.

Exercise 5: Full Actor-Critic Algorithm

- Objective: Simultaneously learn both the value function (critic) and the policy (actor).
- Implementation: The actor-critic algorithm (in actor_critic.py) interleaves updates of both networks based on simulated trajectories.
- Results: Both networks converge toward their optimal counterparts, as indicated by decreasing TD errors and improved policy performance.

Conclusion

All exercises have been implemented successfully. The experimental results verify that the numerical methods and learning algorithms perform as expected. The convergence behavior and comparative analyses support the theoretical foundations outlined in the coursework.