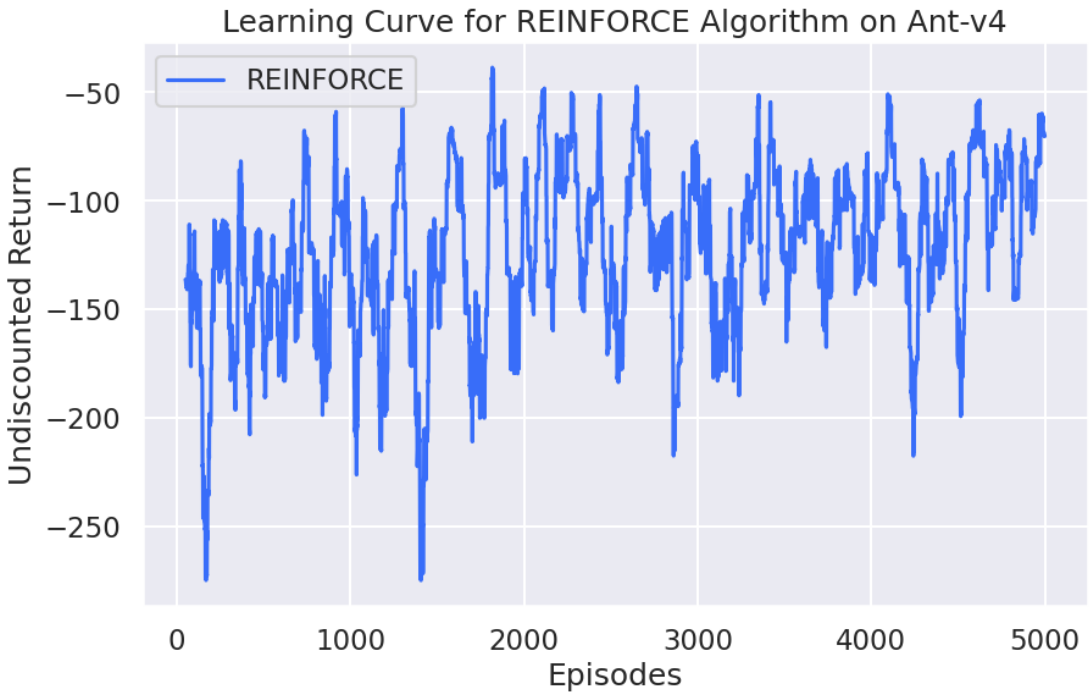
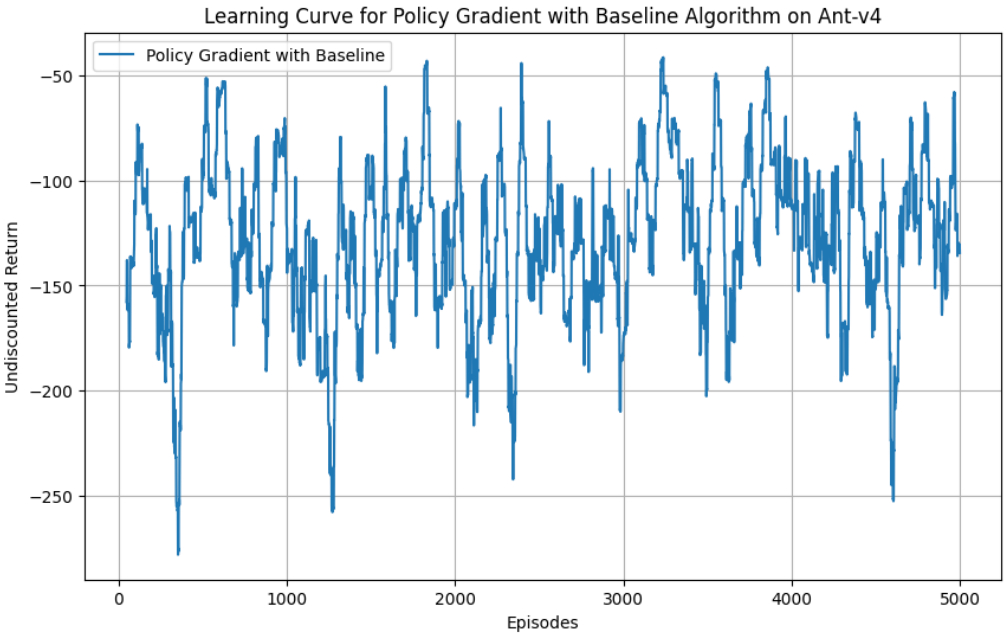
Q2)

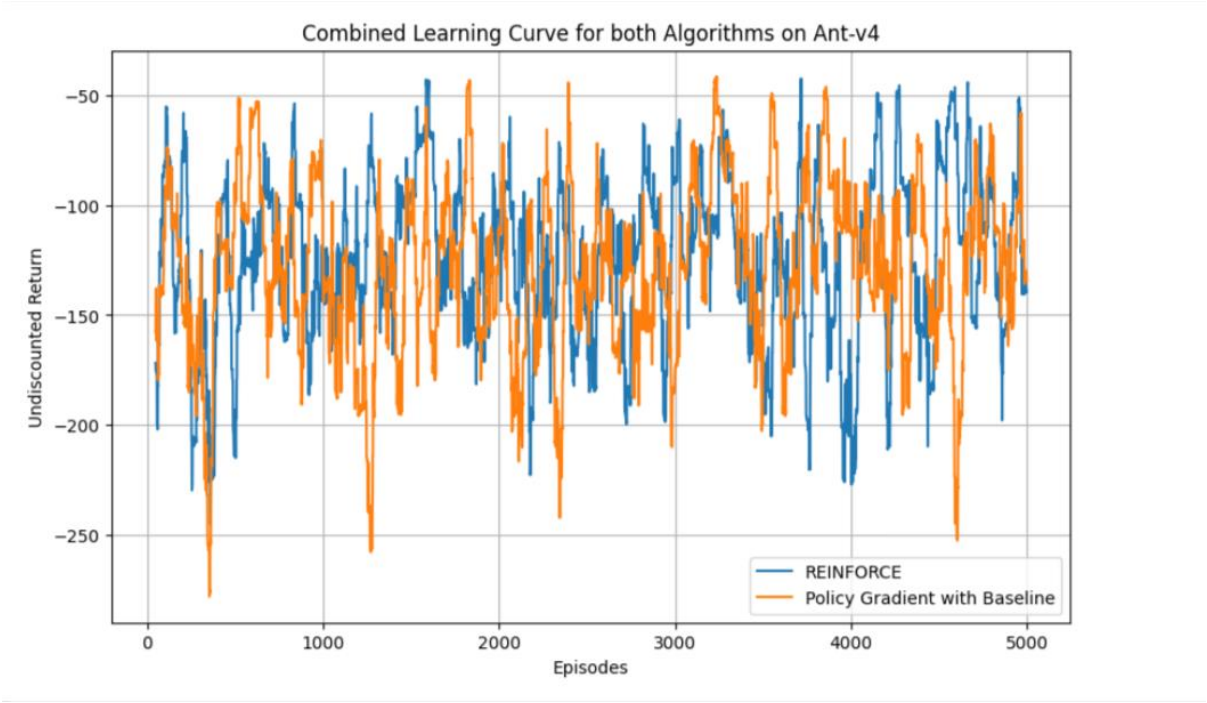
Vanilla Policy Gradient- REINFORCE



Policy Gradient with Baseline



Combined Learning Curve for both Algorithms



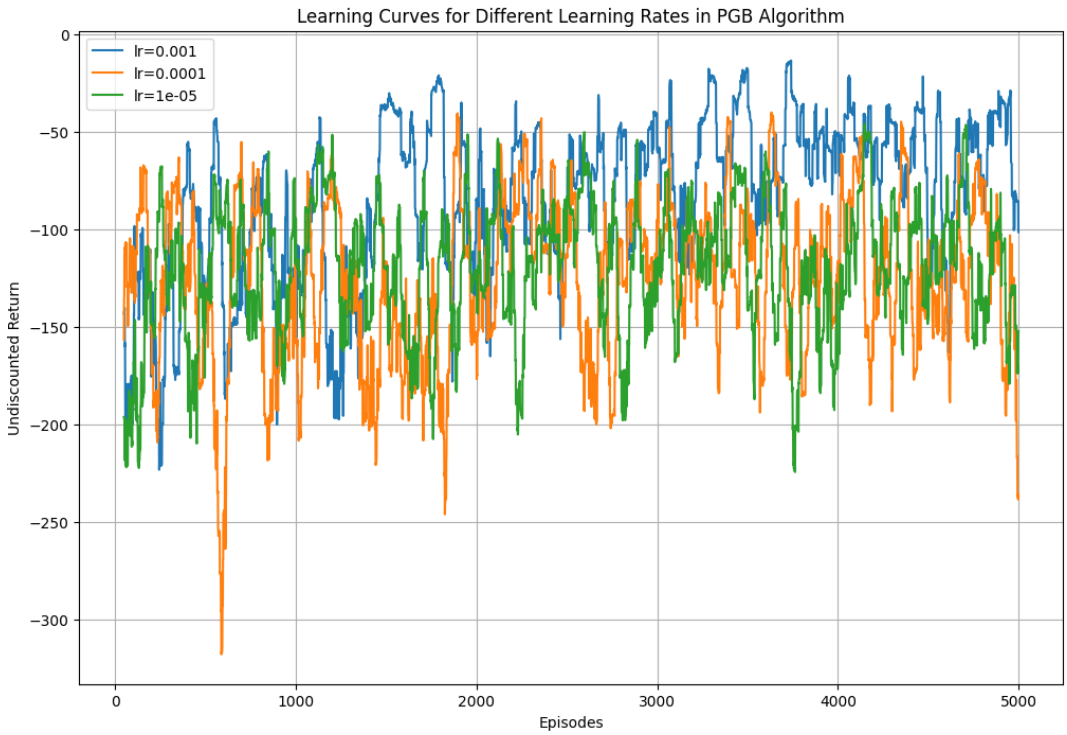
The three graphs represent learning curves for two different reinforcement learning algorithms, Policy Gradient with Baseline and REINFORCE, applied to the same problem domain, Ant-v4. The learning curve tracks the undiscounted return, which is a measure of the cumulative reward received by the agent, over the number of episodes.

The first graph shows a relatively stable performance of the Policy Gradient with Baseline algorithm, with the undiscounted return fluctuating between -50 and -250. The second graph shows the performance of the REINFORCE algorithm, which exhibits a similar range of performance.

Comparing the two, the third graph overlays the learning curves of both algorithms. There is a notable variance in the performance of both algorithms, with neither showing a clear trend towards improvement over time. This could indicate that both algorithms are exploring the policy space and have not yet converged to an optimal policy.

The presence of high variance in the returns suggests that the environment might have a significant amount of stochasticity, or that the policy has not yet stabilized. It's also possible that the parameters for the algorithms need tuning, such as the learning rate, discount factor, or the structure of the policy network. Additionally, the baseline in the Policy Gradient with Baseline algorithm does not seem to provide a significant advantage over the vanilla REINFORCE algorithm in terms of reducing variance or increasing the return, as the performances are quite similar.

Q3)



The graph displays the learning curves for the Policy Gradient with Baseline (PGB) algorithm across 5000 episodes, examining three different learning rates: 0.001 (blue), 0.0001 (orange), and 1e-5 (green). The x-axis represents the number of episodes, while the y-axis shows the undiscounted return per episode.

A few observations can be made:

1. **Fluctuations in Performance**: All three learning rates exhibit significant fluctuations in the undiscounted return, indicating variability in the algorithm's performance from episode to episode. This could be due to the exploration component of the policy or the complexity of the Ant-v4 environment.

2. **Learning Rate Impact:** There is no clear convergence observed for any of the learning rates within the 5000 episodes, which suggests that the task is complex, and the algorithm requires further tuning or a greater number of episodes to learn effectively.

3. **Comparison of Learning Rates**: The learning rate of 0.001 shows the highest and most frequent peaks, which may indicate that while it allows for quicker learning, it might also cause the policy to overshoot the optimal solution. The lower learning rates (0.0001 and 1e-5) exhibit more stable but less pronounced improvements, which could mean that the policy is being updated more conservatively, potentially requiring more episodes to reach optimal performance.

4. **Optimal Learning Rate:** None of the learning rates appear to be clearly superior over the entire span of episodes, suggesting the need for a more nuanced approach, possibly involving learning rate schedules or further hyperparameter optimization.

Overall, the results underscore the challenge of balancing exploration and exploitation, as well as the importance of selecting an appropriate learning rate for stable and efficient training in complex environments like Ant-v4. Fine-tuning other aspects of the learning algorithm, such as the network architecture, discount factor, and update strategy, might also be necessary to achieve better and more consistent performance.