AERO 489

Valasek, Selva

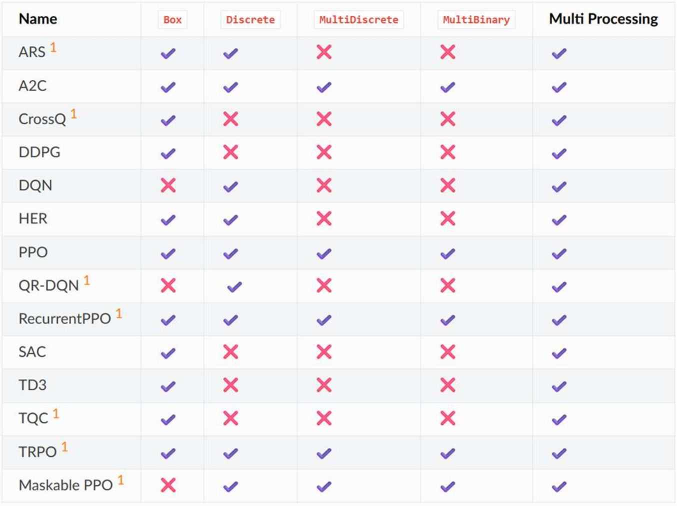
Ian Wilhite

4/13/2025

Acrobot-v1 Machine Learning Hyperparameter Study

Discussion:

1) Gymnasium Environment:
   a. Environment: Acrobot-v1
   b. It is a double pendulum driven from the joint between the arms [1]. The objective is to upright the assembly above the center pivot. This environment has a box observation space and a discrete action space with 3 options for the agent at a given timestep: apply positive torque, apply no torque, apply negative torque [1].
2) Three candidate algorithms:
   a. From the stable baselines compatibility table in Figure 1, we can identify that acrobot requires a discrete action space, meaning that Proximal Policy Optimization (PPO), Advantage Actor Critic (A2C), and Deep Q-Network (DQN) would work with the environment [2]. These algorithms were selected based on their popular usage for a variety of applications and Dr. Valasek's common mention of A2C.



Name	Box	Discrete	MultiDiscrete	MultiBinary	Multi Processing
ARS [1]	✓	✓	✗	✗	✓
A2C	✓	✓	✓	✓	✓
CrossQ [1]	✓	✗	✗	✗	✓
DDPG	✓	✗	✗	✗	✓
DQN	✗	✓	✗	✗	✓
HER	✓	✓	✗	✗	✓
PPO	✓	✓	✓	✓	✓
QR-DQN [1]	✗	✓	✗	✗	✓
RecurrentPPO [1]	✓	✓	✓	✓	✓
SAC	✓	✗	✗	✗	✓
TD3	✓	✗	✗	✗	✓
TQC [1]	✓	✗	✗	✗	✓
TRPO [1]	✓	✓	✓	✓	✓
Maskable PPO [1]	✗	✓	✓	✓	✓

Figure 1. Stable Baselines algorithm environment compatibility

3) Hyperparameter Study:

a. The two hyperparameters I will evaluate are Learning Rate and the Discount Factor. I believe that the learning rate will affect the speed of convergence for the model, as well as the stability of the convergence achieved, whereas the discount factor will determine the performance and aggressiveness of the model in attempting to achieve high performance. An initial investigation can find a workable range for subsequent experiments. Using an average of three trials, a mean can be calculated and plotted in a heat map in Figure 2.

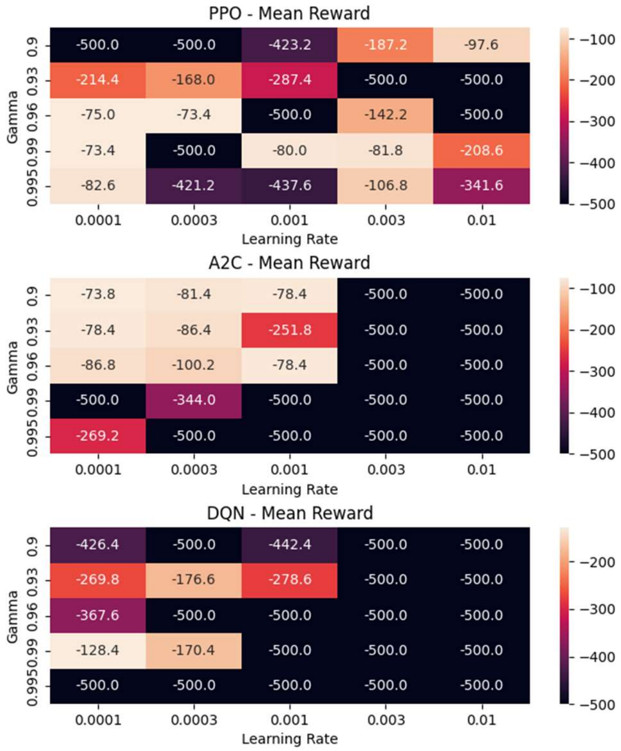


Figure 2. Hyperparameter study for 10,000 episodes

Many of the models simply did not converge, and therefore the training size needed to be increased, however the range seemed promising. It is worth noting that PPO succeeded with little trend, A2C showed a clear successful and unsuccessful region, and DQN showed little progress reinforcing the similar range to A2C.

With an increased timestep, the training curves for each model could be constructed with respect to the discount factor and learning rate. In Figure 4, it can be observed that with lower learning rates (the left columns) that the models converged slower or negligibly, whereas with higher learning rates (the right columns), the models demonstrated oscillation after convergence or suffered failure altogether. The discount factor showed minimal impact, but broadly that a lower discount factor (top rows) created impatient models, that tended to be more aggressive in improving its rewards per episode, whereas a higher discount factor (bottom rows) created a more patient model that converged slower with fewer instances of decreasing rewards between episodes.

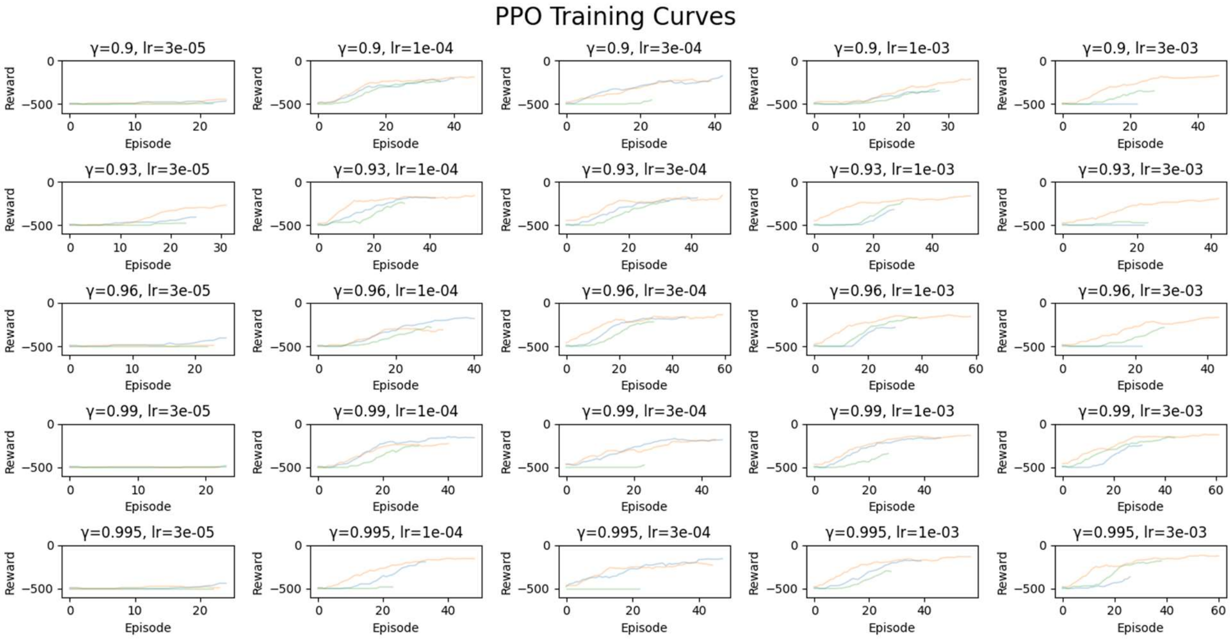


Figure 4. PPO Learning curves for 15,000 episodes w.r.t. discount factor and learning rate

For deep Q learning, all models across the board suffered from increased volatility in convergence. Notably, with very low learning rates they ended training sooner because they converged at negligible rewards.

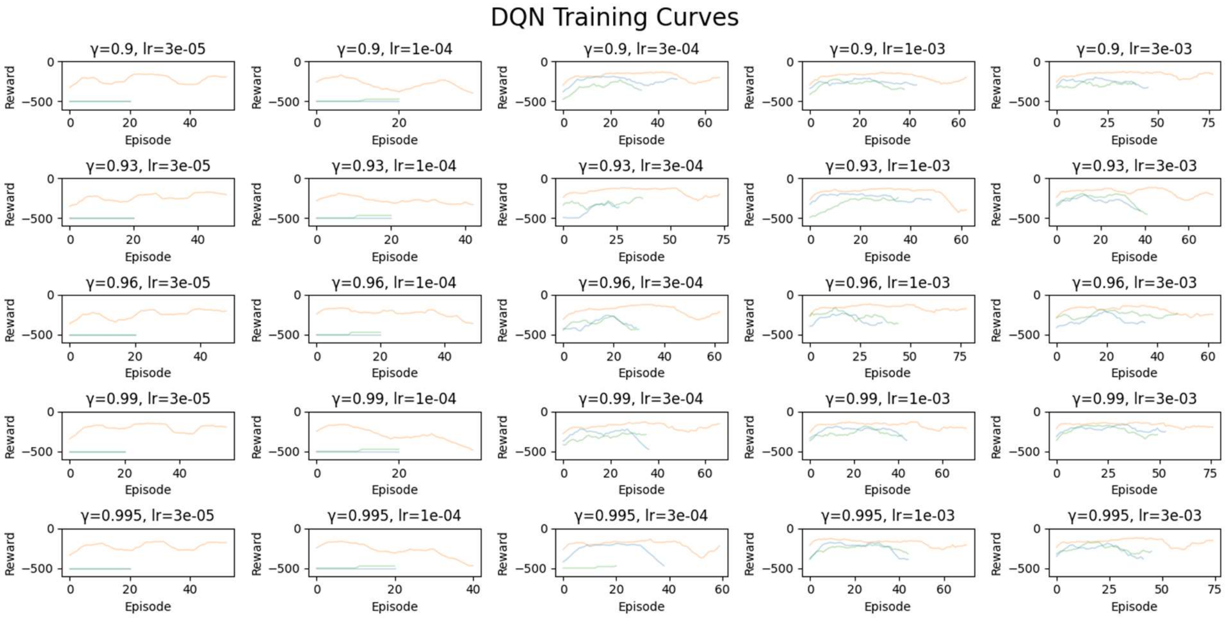


Figure 5. DQN Learning curves for 15,000 episodes w.r.t. discount factor and learning rate

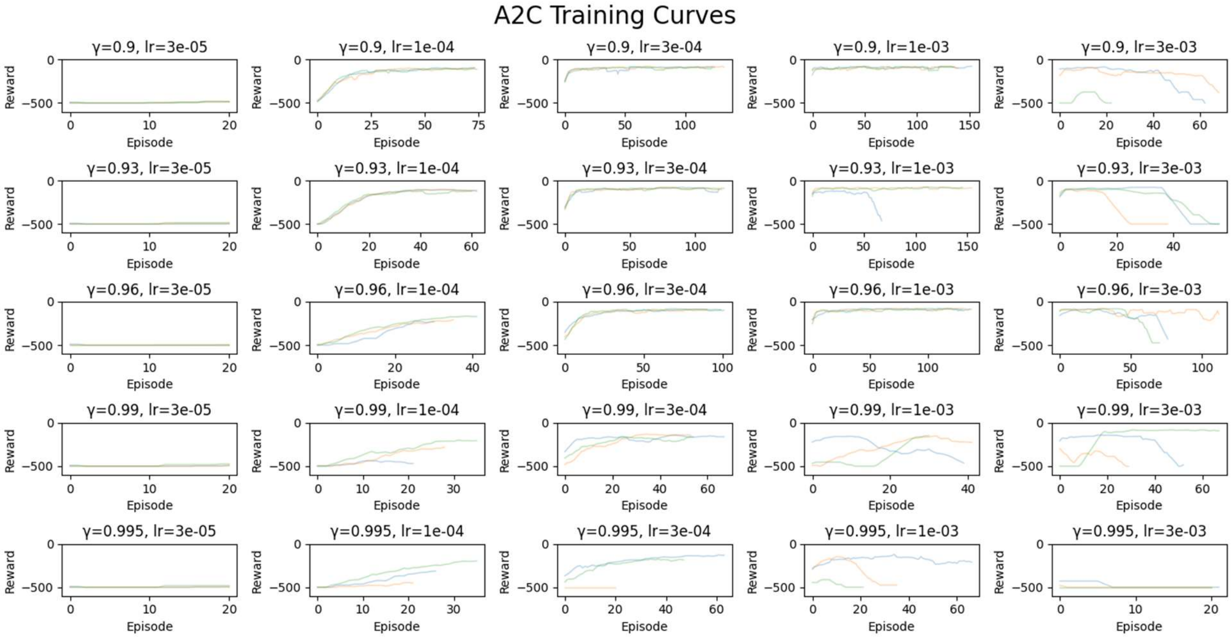


Figure 6. A2C Learning curves for 15,000 episodes w.r.t. discount factor and learning rate

A2C Learning clearly demonstrates the trends stated before, that as the discount factor decreases the model becomes impatient, particularly where the learning rate is effective to converge. When the learning rate is too high, the model does not converge, and when the learning rate is too low, it will not converge or will converge slowly.

4) Direct comparison

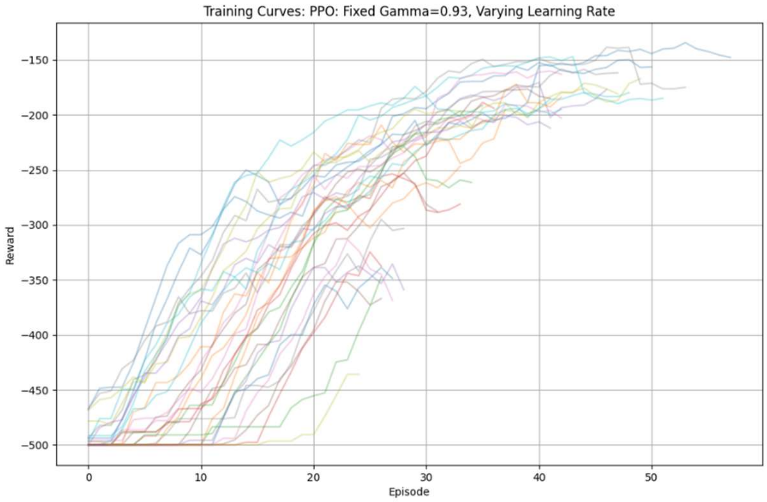


Figure 7. PPO Learning Curves while varying learning rate

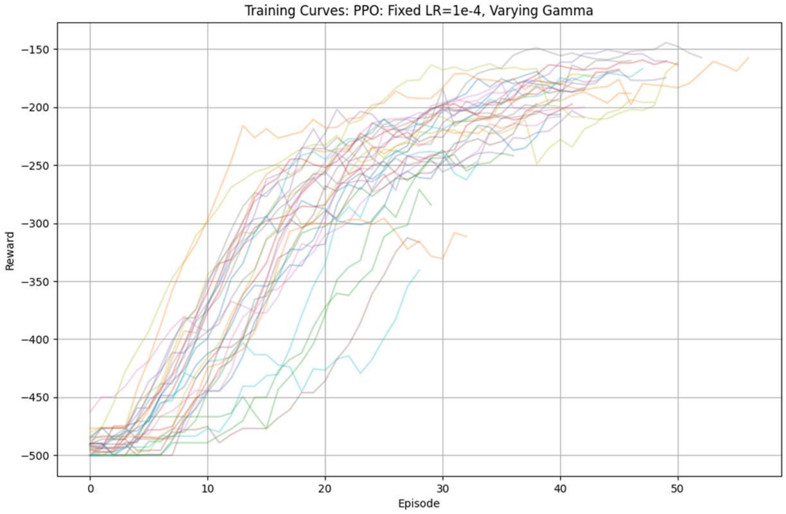


Figure 8. PPO Learning Curves while varying discount factor

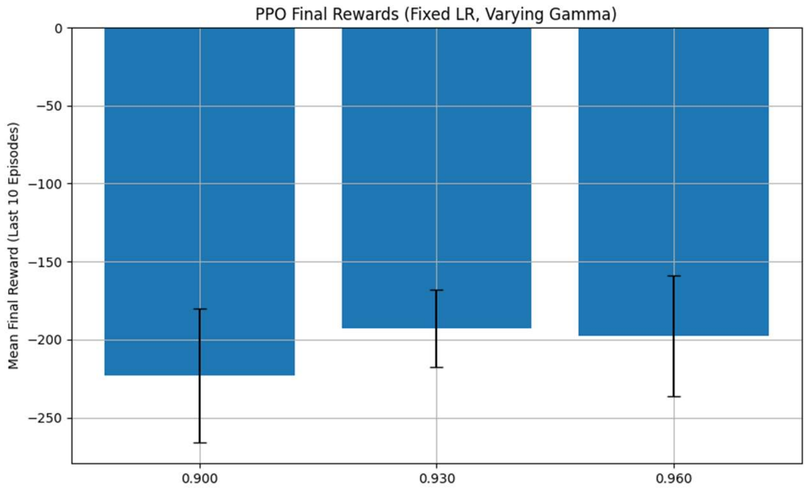


Figure 9. PPO rewards varying discount factor

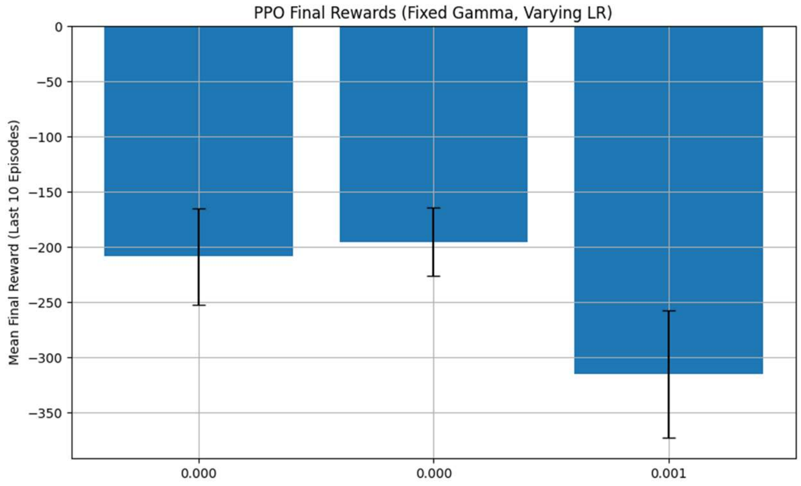


Figure 10. PPO rewards varying learning rate

PPO can be observed as converging consistently, and showing that the lower discount factors decrease the mean resulting reward. It should be noted that the ending rewards are not indicative of the peak rewards or the time to reach peak performance, only that after many iterations

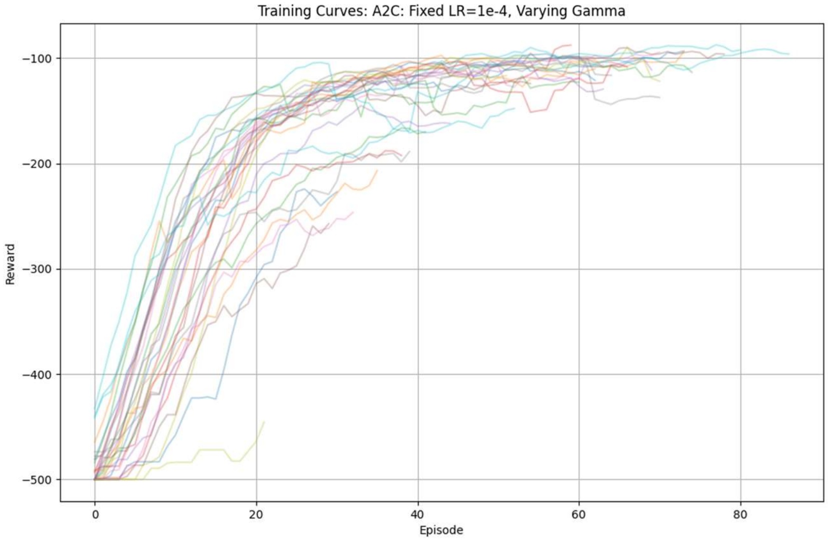


Figure 11. A2C Learning Curves while varying discount factor

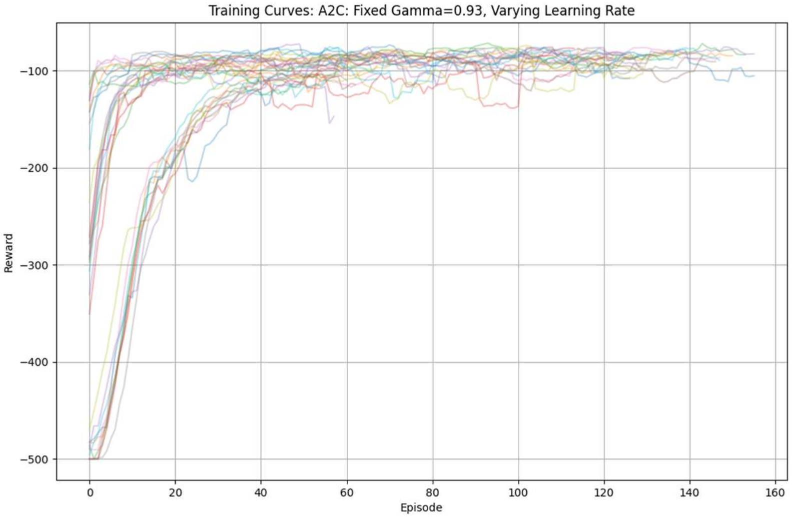


Figure 12. A2C Learning Curves while varying learning rate

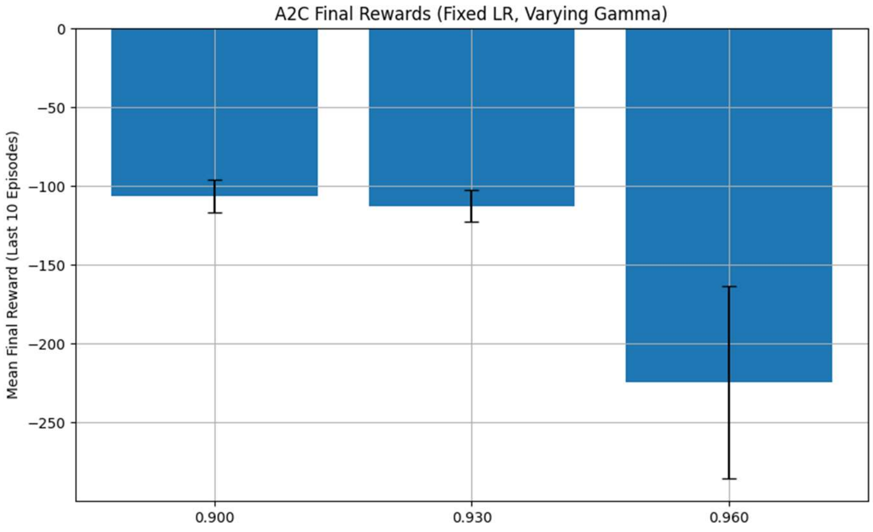


Figure 13. A2C rewards varying discount factor

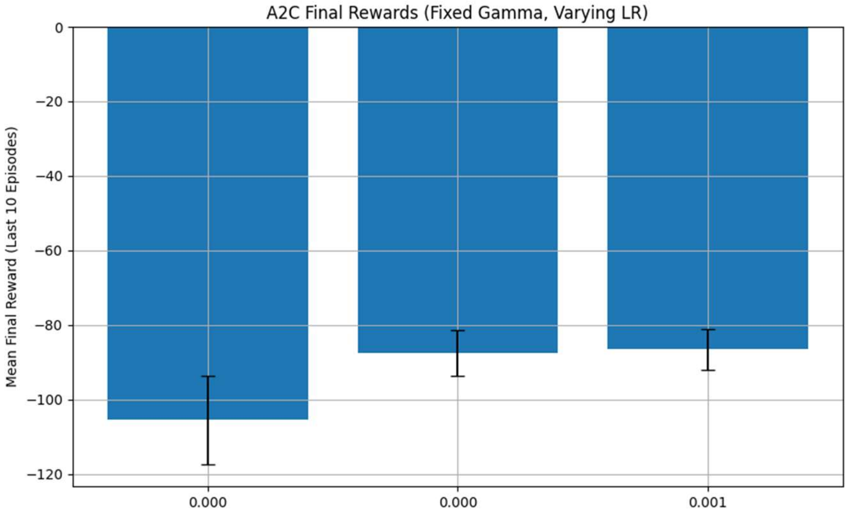


Figure 14. A2C rewards varying learning rate

A2C can be seen as noticeably continuing many of the otherwise observed trends with a lower uncertainty and consistency not only in the final results but also the learning curves path to convergence. The differing learning curves clearly show the paths that each factor takes to convergence. Noting the scale that nearly all of the A2C models outperform the PPO models on similar metrics.

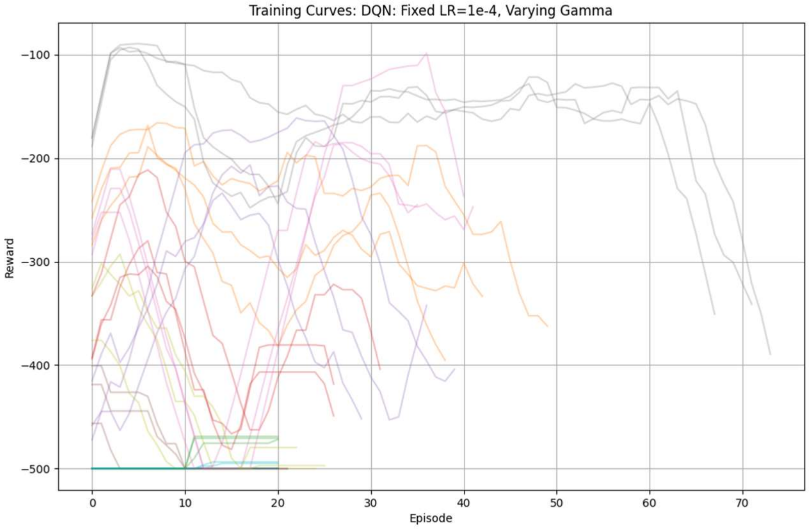


Figure 15. DQN Learning Curves while varying discount

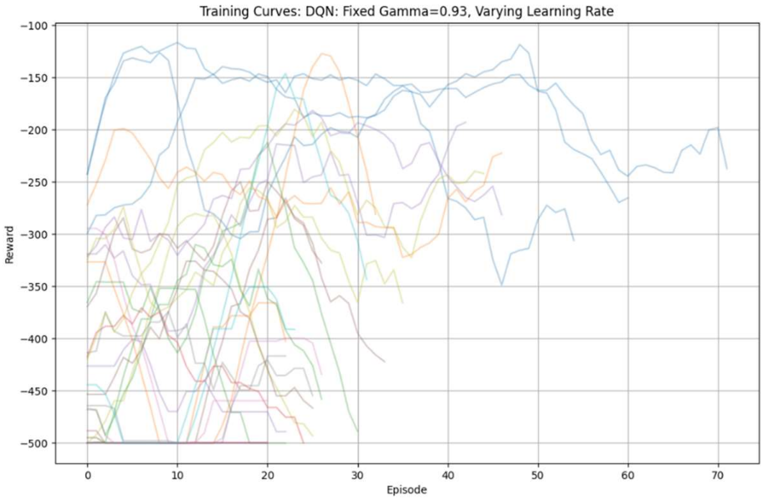


Figure 16. DQN Learning Curves while varying learning rate

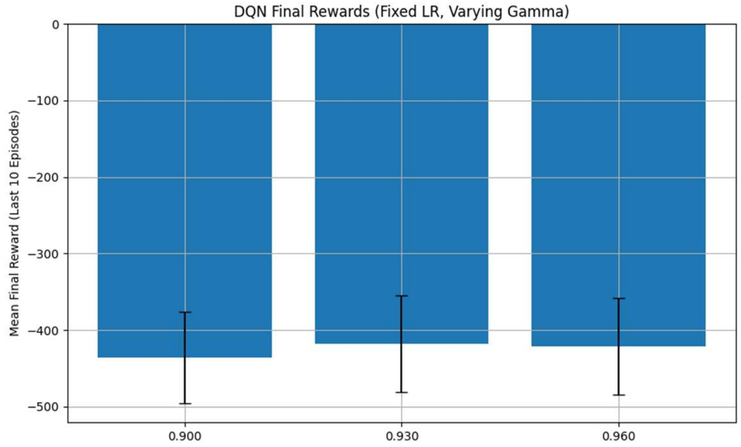


Figure 17. DQN rewards varying discount factor

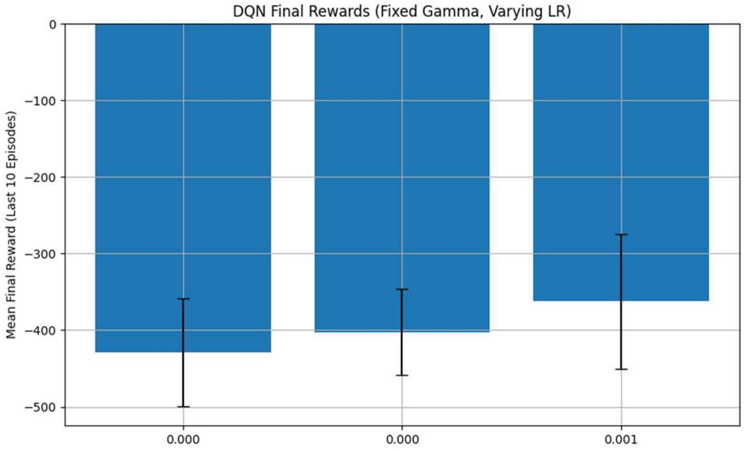


Figure 18. DQN rewards varying learning rate

While PPO demonstrated consistent convergence behavior, A2C outperformed it in both average final rewards and consistency across trials. DQN exhibited unstable learning and failed to converge in most settings, although a few isolated cases approached competitive performance.

5) Conclusion

This study evaluated the impact learning rate and discount factor on PPO, A2C, DQN algorithms using the Acrobot-v1 environment. Results showed that: Low learning rate led to non-convergence, whereas high learning rate led to oscillatory behavior; Low discount factor led to aggressive models that converged quickly while high discount factors created more patient models that converged slower but rarely took any steps that decreased rewards.

a. A2C consistently outperformed PPO and DQN in similar ranges, however it is worth exploring a wider range of learning rates and discount factors as trends for optimality indicated that different models preferred different learning rates and discount factors. PPO could benefit from higher discount factors, A2C could benefit from lower discount factors, and DQN had little impact from discount factors. PPO could benefit from a lower learning rate, whereas A2C and DQN could benefit from a higher learning rate.
b. A2C was the easiest to train, followed by PPO, and I could not get DQN to reliably converge. A2C demonstrated the most consistent convergent given the same hyperparameters.
c. PPO seemed the easiest to tune in the sense that poor hyperparameters could be accounted for with additional training episodes. A2C had a very clear optimal region of performance with respect to its hyperparameters, meaning that although it was harder to find this optimal region of performance, inside there was consistent and rapidly converging high performance. DQN did not converge consistently across any region of hyperparameters and was consistently exiting training episodes, indicating a need for continued variance in the hyperparameters provided.
d. This is the most I have worked with machine learning models thus far, and I think it was an interesting way to better understand the role that hyperparameters can play in the training process for various models. I learned about the characteristics of PPO, A2C, and DQN, and the impacts that learning rate and discount factor have on the performance of various algorithms. I think that this has been one of the most beneficial assignments I have seen in the class so far with respect to applications of the concepts that we have been learning up to this point

Future work could include a broader range of learning rates and discount factors applied, increased trials to decrease uncertainty and determine trends more consistently, and optimization techniques applied to identify optimal parameters for different algorithms.

Appendix

[1] Farama Foundation. *Acrobot-v1 Environment Documentation*. Gymnasium Classic Control Environments. Available at: https://gymnasium.farama.org/environments/classic_control/acrobot/. Accessed April 13, 2025.

[2] Stable-Baselines3 Developers. *Algorithm Guide and Environment Compatibility*. Stable-Baselines3 Documentation. Available at: https://stable-baselines3.readthedocs.io/en/master/guide/algos.html. Accessed April 13, 2025.