CS 667 Final Project: Solving HumanoidStandup-v5 Using the Soft Actor Critic Reinforcement Learning Method

Written by Blake Crockett

326 Rose St, MDS Room 220
Lexington, KY 40508 USA
bdcr235@uky.edu

Abstract

This final project implements a modified version of the Soft Actor Critic model from Stable Baselines 3 to solve the HumanoidStandup-v5 environment. This environment is a part of Open AI's Gymnasium library. The goal of this environment is for the agent to learn how to orient itself in an upright standing position when starting from lying on its back. A reward was earned at each time step during an episode based on an upward movement incentive, a control cost penalty for large actions, and an impact cost penalty for excessive external forces, balancing performance and stability in humanoid motion. The performance of the model was evaluated by its total episode reward gained over time, and its average cumulative reward across 100 testing episodes.

HumanoidStandup-v5 Environment — https://gymnasium.farama.org/environments/mujoco/humanoid_standup/

Stable Baselines 3 Repository — https://github.com/DLR-RM/stable-baselines3

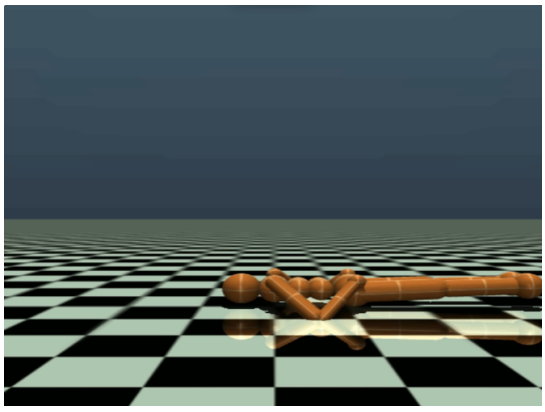

Figure 1: Example Starting Position for HumanoidStandup-v5 Environment.

Introduction

This final project concludes CS 667-001 (Sequential Decision Making), instructed by Dr. Brent Harrison at the University of Kentucky, by implementing an advanced reinforcement learning algorithm to solve a sequential decision-making challenge. Sequential decision-making is a critical focus in computer science, influencing applications in autonomous technologies, robotics, recommendation systems, and strategy-based games. Reinforcement learning (RL) is integral to this domain, as it allows agents to derive optimal policies through dynamic interaction with their environment.

In this assignment, the Soft Actor Critic algorithm was implemented to solve the HumanoidStandup-v5 environment provided by OpenAI's Gymnasium. Soft Actor Critic is a powerful reinforcement learning method designed for continuous control tasks, offering sample-efficient learning through off-policy updates and entropy maximization. Its twin Q-networks and stochastic actor-critic structure provide reliable value estimates and promote consistent exploration. These features make Soft Actor Critic particularly effective for addressing the complexities of the HumanoidStandup-v5 environment, where precise control of a humanoid agent is required to achieve an upright position.

The HumanoidStandup-v5 environment in Gymnasium, powered by the MuJoCo physics engine, involves teaching a humanoid robot to stand up from a resting position. This environment is significantly more complex than classic control tasks due to its high-dimensional state and continuous action spaces. The agent must coordinate numerous joints while managing stability and external forces. Soft Actor Critic, with its entropy-regularized framework and robust value estimation through twin Q-networks, is well-suited for such dynamic tasks. The reward structure encourages upward movement while applying penalties for excessive force and instability.

This report presents the implementation of Soft Actor Critic for the HumanoidStandup-v5 task, discussing the necessary adjustments to adapt the algorithm to a high-dimensional MuJoCo environment. The methods, results, and discussions provide insights into the challenges of achieving optimal performance, focusing on learning efficiency and stability.

Methods

HumanoidStandup-v5 Environment

The HumanoidStandup-v5 environment, part of OpenAI Gymnasium and powered by MuJoCo, involves controlling a humanoid robot to stand up from a supine position. The robot comprises a torso and limbs with multiple joints, and the agent must learn to apply joint torques effectively to achieve the task.

- **Action Space:** A 17-dimensional continuous action space representing torques applied to the robot's joints.
- **Observation Space:** High-dimensional, including joint positions, velocities, and external forces.

Reward Function: The total reward consists of three key components:

$$r_{\text{total}} = \text{uph\_cost} + 1 - \text{quad\_ctrl\_cost} - \text{impact\_cost},$$

where:

1. **Uphill Reward (uph_cost)**: Incentivizes upward movement of the torso:
$$\text{uph\_cost} = w_{\text{uph}} \times \frac{z_{\text{afteraction}} - z_{\text{beforeaction}}}{dt},$$
where $z_{\text{afteraction}}$ is the torso's z-coordinate after the action, dt is the time step (default 0.05), and w_{uph} is a weight factor.
2. **Control Cost (quad_ctrl_cost)**: Penalizes large torque values applied to the joints:
$$\text{quad\_ctrl\_cost} = w_{\text{ctrl}} \times \|\mathbf{a}\|_2^2,$$
where $\mathbf{a}$ is the vector of applied torques and w_{ctrl} is a control weight factor.
3. **Impact Cost (impact_cost)**: Penalizes excessive external contact forces:
$$\text{impact\_cost} = w_{\text{impact}} \times \text{clamp}(\text{range}, \|F_{\text{contact}}\|_2^2),$$
where $\|F_{\text{contact}}\|_2^2$ is the squared L2 norm of contact forces, and w_{impact} is a weight factor.

Objective: The goal is to maximize the cumulative reward by learning to efficiently control the humanoid's joints to achieve a standing position while minimizing penalties.

Soft Actor-Critic

Soft Actor-Critic is an off-policy, model-free reinforcement learning algorithm designed for continuous control tasks. Soft Actor Critic maximizes a trade-off between expected return and policy entropy, encouraging exploration and robustness. The key components are:

- **Entropy Regularization:** The objective includes an entropy term to balance exploration:
$$J_\pi = E\left[\sum_{t=0}^{T} \gamma^t (r_t + \alpha \mathcal{H}(\pi(\cdot|s_t)))\right],$$
where α controls entropy weight.
- **Twin Q-Networks:** Two Q-networks Q_1 and Q_2 reduce overestimation bias, with target Q-values given by:
$$y = r + \gamma \left(\min_{i=1,2} Q_i(s', a') - \alpha \log \pi(a'|s')\right).$$
- **Stochastic Policy:** The actor samples actions from a squashed Gaussian distribution:
$$a = \tanh(\mu(s) + \sigma(s) \cdot \epsilon), \quad \epsilon \sim \mathcal{N}(0, 1).$$

Soft Actor Critic Implementation in Stable Baselines3

Stable Baselines3 implements Soft Actor Critic for efficiency and numerical stability. Key implementation highlights include:

- **Actor Network:** Outputs actions using a Gaussian distribution, with log standard deviation capped between -20 and 2 to avoid instability.
- **Twin Q-Networks:** Critic networks estimate Q-values, with a target network updated via Polyak averaging for stability.
- **Entropy Coefficient:** The entropy weight α can be fixed or learned automatically to match a target entropy.
- **Replay Buffer:** Soft Actor Critic uses an off-policy replay buffer for efficient learning from stored experiences.

The networks are updated using gradient descent, ensuring stable and sample-efficient learning for continuous action spaces. Hyperparameters used for training are shown in Table 1.

Hyperparameter	Value
Learning Rate	3e-4
Discount Factor (γ)	0.99
Tau (τ)	0.005
Batch Size	245
Gradient Steps	1
Buffer Size	1e6
Entropy Coefficient	auto
Learning Starts	1e4
Temperature (T)	0.2

Table 1: Hyperparameters used for Soft Actor Critic.

Training Details

The Soft Actor Critic agent was trained over **25,000,000 time steps** (equivalent to **25,000 episodes**), which required **over 48 hours** of continuous computation. Testing was performed on **100 episodes** to evaluate the stability and generalization of the learned policy. The extended training duration reflects the complexity of the HumanoidStandup-v5 task and the need for precise control in high-dimensional action spaces.

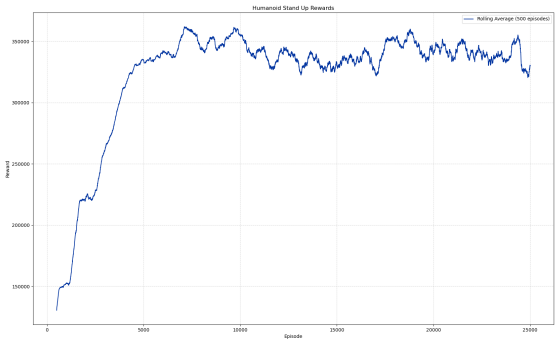

Figure 2: Humanoid in Standing Position After Training.

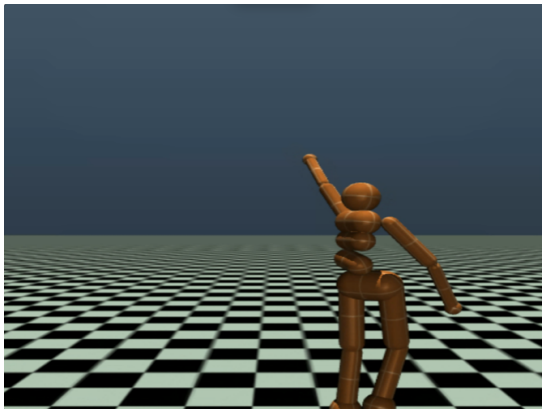

Figure 3: Humanoid in Standing Position After Training.

Results and Analysis

The training results for the HumanoidStandup-v5 environment are presented in Figure 2. The graph shows the rolling average rewards (over 500 episodes) achieved by the Soft Actor Critic agent during the 25,000 training episodes.

Performance Trends: Initially, the cumulative rewards increased steadily as the agent learned to achieve upward movement, with rapid improvements during the first 5,000 episodes. The rewards plateaued between episodes 7,000 and 10,000, indicating that the agent achieved consistent standing behavior. After this stage, fluctuations in the rewards are observed, which likely result from exploration dynamics and entropy regularization decay. A maximum reward of 401421.9 was achieved during training.

Convergence and Stability: The agent stabilized around an average reward of approximately 350,000, showcasing the Soft Actor Critic algorithm's ability to handle the high-dimensional continuous control task. However, minor instabilities persisted toward the end of training, suggesting further tuning (e.g., entropy coefficients or learning rate adjustments) could smooth the convergence.

Testing Results: The policy was evaluated over 100 test episodes, demonstrating consistent upward movement and standing performance, aligning with the training results. The average cumulative reward over the course of the 100 episodes was 382,354.77 with a standard deviation of $\pm$ 60,560.60. The results confirm the effectiveness of Soft Actor Critic in learning robust policies for complex environments.

Discussion

HumanoidStandup-v5 Environment

The HumanoidStandup-v5 environment presented a significant challenge due to its high-dimensional continuous action space and complex joint dynamics. Training the Soft Actor Critic algorithm on this environment required careful tuning and extended computation. Initial runs revealed instability in rewards, which was mitigated through entropy regularization and gradient clipping.

A critical decision was to train the model for **25,000,000 time steps** (approximately **25,000 episodes**), which took over 48 hours to complete. The rolling average reward steadily improved, stabilizing around 350,000 (Figure 2). However, fluctuations persisted, indicating the ongoing impact of entropy decay and exploration during later stages of training.

Initial attempts at training the model resulted in early convergence to a local maximum. In these situations, the humanoid could learn to sit upright but failed to use its legs to get itself off the ground. Tuning the hyperparameters was costly, due to the long training times. Often, it would take hours to conclude that the model had gotten stuck. Three key changes to the model were the direct cause of the successful training detailed previously. They are:

- **Reduced State Space:** The base observation space of 348 unique values likely caused unwanted noise during training. Lowering it to 45 measurements of position and velocity of the body parts allowed the model to better find meaningful updates.
- **Auto Entropy Regularization:** Other reinforcement learning models, like Proximal Policy Optimization, use a static entropy coefficient. Carrying this idea over to Soft Actor Critic does not allow for an appropriate level of exploration in the environment before converging on a solution.
- **More Frequent Updates:** While training was much quicker with sparsed out updates, the results were not optimal. Training every step was a main factor in the higher rewards.

Performance testing on **100 episodes** confirmed the agent's ability to achieve consistent standing behavior, with cumulative rewards aligning closely with the observed training trends. The high-dimensional control task showcased Soft Actor Critic's strengths in handling complex environments, albeit at a significant computational cost.

Comparative Analysis

Compared to simpler environments like CartPole or Lunar Lander, the HumanoidStandup-v5 task highlights the scalability and robustness of Soft Actor Critic for continuous

control problems. Unlike discrete action tasks, where simpler architectures suffice, this environment required a more advanced approach to stabilize learning. The large action space and reward sparsity made adaptive techniques, such as learning rate scheduling and entropy decay, essential for success.

The results demonstrate that Soft Actor Critic is well-suited for environments with complex dynamics, such as the HumanoidStandup-v5 task. However, achieving optimal performance in such environments requires significant computational resources and careful hyperparameter tuning to ensure stability and convergence.

Acknowledgments

I would like to express my sincere gratitude to Dr. Brent Harrison for providing an in-depth and practical course on sequential decision-making and reinforcement learning, which formed the foundation of this project. His emphasis on both theoretical concepts and their practical applications greatly enhanced my understanding of reinforcement learning algorithms.

I also acknowledge the use of OpenAI's GPT-4 model as a resource during this assignment. A large barrier for this project was learning to get MuJoCo to run and render on my Windows machine. ChatGPT was able to help in conjunction with numerous online forums.

Finally, I would like to thank the Gymnasium and MuJoCo teams for providing the Humanoid Standup environment. Their robust simulation frameworks allowed me to explore and evaluate the capabilities of the Soft Actor Critic algorithm in a high-dimensional continuous control task.