CS-GY 6923 Machine Learning Project Proposal - Fall 2025

Group Members: - Ansh Harjai **(ah7163)**
- Apoorva Menon **(as22037)**
- Shyam Krishna Sateesh **(ss20355)**

Project Title: Multi-Agent and Stochastic Reinforcement Learning in an Extended Taxi Environment

1. Problem Statement

We aim to train autonomous agents to efficiently manage passenger pick-ups and drop-offs in an extended version of the Taxi-v3 environment. The standard Taxi-v3 problem (a single agent in a deterministic world) is a solved benchmark. However, it fails to capture the complexity of real-world logistics, which involve **multiple, competing agents** and **environmental uncertainty**.

Our project will extend this classic environment to explore these more difficult challenges. We will investigate:

- **Multi-Agent Dynamics (MARL):** How do multiple taxi agents learn to operate in the same grid? We will model both **competitive** scenarios (e.g., agents race for the same passenger) and **cooperative** scenarios (e.g., agents are judged by a total, shared reward).
- **Stochasticity:** How do optimal policies change when the environment is no-longer deterministic? We will introduce a "stochastic-slip" probability, where an agent's intended action (e.g., 'move north') has a small chance of failing (e.g., moving east or west instead).

This creates a more realistic and difficult problem that is a well-known challenge in modern reinforcement learning.

2. Dataset

No external dataset is required for this project. All data is generated through interactions between the agent and the Taxi-v3 simulation environment. The environment is fully structured and requires no preprocessing.

Our project will leverage two core libraries from the Farama Foundation:

a. **Gymnasium (Single-Agent):** We will use the standard `gymnasium.make("Taxi-v3")` for our single-agent baselines. We will also use a Gymnasium "wrapper" to introduce stochasticity.

b. **PettingZoo (Multi-Agent):** For the multi-agent version, we will develop a custom parallel environment that follows the **PettingZoo API**. PettingZoo is the standard library for MARL and the sister library to Gymnasium.

The base environment properties are:

- **Action Space:** Discrete (6) - south, north, east, west, pickup, dropoff.
- **Observation Space (Base):** Discrete (500) - 25 taxi positions, 5 passenger locations, 4 destinations.
- **Rewards (Base):** +20 (success), -1 (per step), -10 (illegal action).

Our PettingZoo implementation will extend this by:

- Defining a joint state space that includes the locations of all agents.
- Handling a `step()` function that receives a dictionary of actions from all agents simultaneously.
- Implementing competitive and cooperative reward structures.

3. Models / Algorithms

We will implement a tiered comparison of algorithms, where each tier addresses a different aspect of our problem:

- Baseline (Single-Agent): Standard Q-learning and SARSA on the deterministic Gymnasium Taxi-v3 environment.

- Stochastic Baseline (Single-Agent): We will re-run Q-learning and SARSA on our stochastic Gymnasium wrapper to analyze the policy change.

- Multi-Agent Learning (MARL): Using our new PettingZoo environment, we will implement Independent Q-Learning (IQL). This is a common MARL baseline where each agent learns its own Q-function, treating other agents as part of the environment.

- Advanced Model (MARL): The multi-agent state space is combinatorially larger. To handle this, we will implement a Deep Q-Network (DQN)-based approach (e.g., Independent DQN or DQN with parameter sharing), which is compatible with the PettingZoo API.

4. Evaluation Metrics

Performance will be evaluated using:

- Average reward per episode (per-agent and system-wide)
- Number of steps to complete an episode
- Success rate of drop-offs

- Convergence rate of learning (e.g., Q-value stability)
- System-wide Efficiency: Total passengers delivered by all agents per 1000 timesteps.
- Policy Analysis (Qualitative): We will visualize the learned policies to identify emergent behaviors (e.g., map-splitting, agent-blocking).

5. Previous Work and Improvements

Taxi-v3 is a standard RL benchmark, and its single-agent solution is well-documented. Our project's primary contribution is to use this classic environment as a testbed for investigating more complex, graduate-level research questions.

Our improvements on previous work will be:

- Standard-Compliant Environment: We will develop and open-source a multi-agent, stochastic version of the Taxi environment that adheres to the PettingZoo API, making it a reusable testbed for other researchers.

- Comparative MARL Analysis: We will provide a rigorous analysis of how different MARL strategies (IQL vs. DQN) perform in both competitive and cooperative settings.

- Stochastic vs. Deterministic Policy: We will analyze how stochasticity fundamentally alters the learned optimal policies compared to the deterministic baseline.