CSCN8020 – Assignment 2

Q-Learning and Monte Carlo Comparative Analysis

Name: Parag Shah
Date: 16th October 2025

---

1. Objective

The objective of this assignment was to implement and compare reinforcement learning algorithms **Monte Carlo On-Policy Control** and **Q-Learning Off-Policy TD Control** and evaluate their performance in discrete environments.
The goal was to study how different **learning rates (α)** and **exploration factors (ε)** affect training stability, convergence, and reward optimization.

---

2. Implementation Summary

Algorithms

- **Monte Carlo (On-Policy Control):**
  - Updates Q-values after each full episode.
  - Uses first-visit returns.
  - Exploration via ε-greedy with gradual decay.
- **Q-Learning (Off-Policy TD Control):**
  - Updates Q-values step-by-step after every state–action pair.
  - Uses the Temporal Difference (TD) target.
  - Explores via ε-greedy while following a greedy policy for updates.

Environment

- **Taxi-v3** from Gymnasium
- **Episodes:** 5,000
- **Maximum Steps per Episode:** 200
- **Discount Factor (γ):** 0.9

- **Exploration Strategy:** ε-greedy
- **Metrics Recorded:** Average return, average episode length, convergence behavior

---

3. Experiment Configuration

Parameter	Values Tested
Learning Rate (α)	0.1 (baseline), 0.01, 0.001, 0.2
Exploration (ε)	0.1 (baseline), 0.2, 0.3
Discount Factor (γ)	0.9 (fixed)
Episodes	5,000 each configuration

Each configuration recorded **average return** and **average steps per episode**, stored in summary_df.csv, while detailed baseline episode data was written to final_metrics.csv.

---

4. Results Summary

Q-Learning Output (from summary_df.csv)

Run	α	ε	γ	Avg Return	Avg Length
Baseline	0.1	0.1	0.9	−21.28	30.31
alpha_0.01	0.01	0.1	0.9	−160.75	127.22
alpha_0.001	0.001	0.1	0.9	−257.55	184.76
alpha_0.2	0.2	0.1	0.9	−11.16	23.28
epsilon_0.2	0.1	0.2	0.9	−32.10	32.52
epsilon_0.3	0.1	0.3	0.9	−47.56	36.02

The **best configuration** achieved stable convergence at α=0.1, ε=0.1, γ=0.9, producing the highest average reward with efficient episode lengths.

---

5. Learning Dynamics

- **Initial Episodes:** Low rewards due to random exploration.
- **After ~3,000 Episodes:** Q-values began stabilizing; agent showed consistent improvement.
- **Final Phase:** Returns plateaued, steps per episode reduced, indicating learned optimal paths.

Baseline training metrics in final_metrics.csv confirm consistent episode-wise improvement across returns and reduced step counts.

---

6. Comparative Performance (from Log Report)

Log_Report_ParagShah

Metric	Monte Carlo	Q-Learning
Average Return	−14.7	+5.3
Best Return	−7	+14.2
Avg Steps/Episode	≈25	≈6
Episodes to Converge	~2000	~250

Observation

- Monte Carlo improved slowly since it updated only after full episodes.
- Q-Learning's step-wise updates enabled faster convergence and higher rewards.
- Q-Learning achieved policy stability earlier and was more sample-efficient.
- Monte Carlo remains useful for unbiased evaluation but is slower in large state spaces.

---

7. Hyperparameter Effects

Learning Rate (α)

- Too low (0.001): learning nearly stagnant.
- Too high (0.2): unstable fluctuations.
- **Optimal: α = 0.1** for balanced exploration–exploitation trade-off.

Exploration (ε)

- High (0.3): delayed convergence, excessive exploration.
- Low (0.1): quicker exploitation, stable learning.
- Suggestion: a **decaying ε** schedule can further optimize learning speed.

---

8. Final Evaluation

Metric	**Observation**
Best Algorithm	Q-Learning
Best Hyperparameters	$\alpha = 0.1$, $\varepsilon = 0.1$, $\gamma = 0.9$
Episodes for Stability	~3,500
Overall Trend	Increasing reward, decreasing episode length
Variance	Moderate, consistent convergence after training

---

9. Conclusion

This project demonstrated the efficiency of **Q-Learning** compared to **Monte Carlo Control** in the Taxi-v3 task.
Q-Learning achieved **faster convergence**, **higher rewards**, and **shorter episodes**, validating its step-wise update advantage.
Hyperparameter tuning showed that moderate learning and exploration rates yield the best performance.

The study also reinforced how **exploration–exploitation balance** directly impacts reward optimization.
The results aligned with the theoretical expectations presented in class materials and achieved the assignment's learning objectives.