Implementing Reinforcement Learning on Continuous Control Problems

Justin Stephenson, Chloe Dimpas, Neves Soares

Justin Stephenson

Overview

Let’s see what this presentation is all about

Brief Overview

This project/presentation is all about comparing and contrasting different implementations of reinforcement algorithms focusing on the continuous control problem space. We will see how these algorithms perform with different hyperparameters and when increasing the complexity of the problem space (environment).

We will be comparing and contrasting the three following reinforcement algorithms:

- Q-Learning
- SARSA - State Action Reward State Action
- PPO - Proximal Policy Optimization

We will be focusing on two continuous control problems from OpenAI:

- Cart Pole problem
- Mountain Car problem

Neves Soares

Reinforcement Algorithms

What is going on behind the scenes

Reinforcement Algorithms

Q-Learning

An adaptation of the Q-Value Iteration algorithm seen in MDP (Markov Decision Process) where the transition probabilities and rewards are unknown. Works by watching an agent 'play' and improving estimates of Q-values.

SARSA

Variation of Q-Learning algorithm. SARSA technique is on-policy and uses the action performed by the current policy to learn the Q-value. Two consecutive state-action pairs and the immediate reward determine the updated Q-value.

PPO

PPO is a policy gradient method, that trains a stochastic policy in an on-policy technique. It utilizes an actor-critic method, where the actor maps the observation to an action and the critic gives an expectation of the rewards of the agent for the observation given.

Chloe Dimpas

Continuous Control Problems

Thinking about the problem space

Continuous Control Problems

Description:
An environment/problem where at any given time the state of the agent can be an unspecified number of possible measurements between two realistic points.

The solution to deal with continuous control:

These algorithms in particular work best on discrete spaces. This is because the Q-table will be enormous if we consider each state of a continuous control problem. To solve this, we ‘discretized’ each environment, reducing the number of states to a manageable number. This adds an extra hyperparameter we have to choose.

Justin Stephenson

Mountain Car

Get to the top of the mountain ASAP

Mountain Car - Description

Goal: Get the car to the checkpoint indicated by the flagpole. Do this before 200 episode iterations elapse.

Actions:

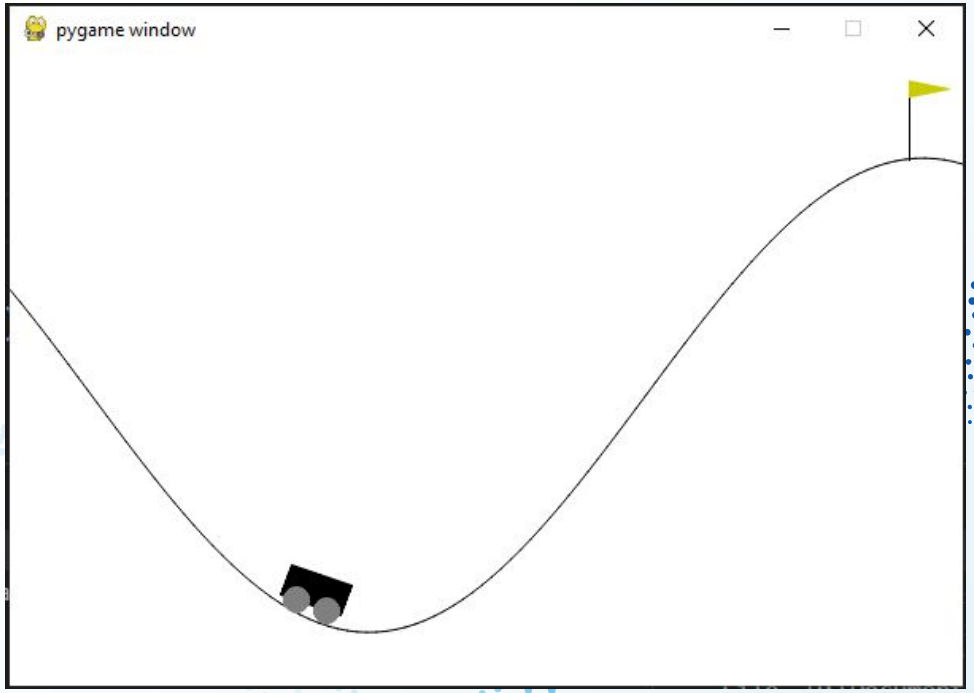
- Accelerate to the left
- Don’t accelerate
- Accelerate to the right

States: A combination of:

- Position of car on x-axis
- Velocity of car

Rewards:

- -1 for every episode iteration
- 0 if goal is reached



Mountain Car - Compare and Contrast (Standard)

Hyperparameters:

Episodes = 50,000

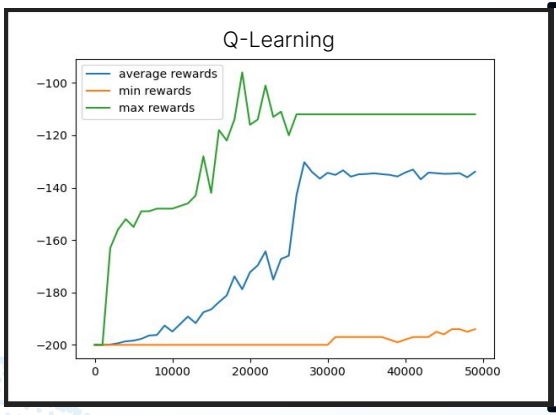
Episode Iterations = 200

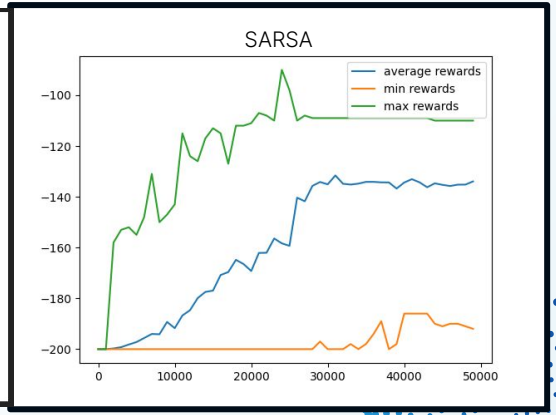
Learning Rate = 0.1

Discount = 0.95

Epsilon = 0.5

Discrete States = 400





Mountain Car - Compare and Contrast (Learning Rate Increase - Factor of 5)

Hyperparameters:

Episodes = 50,000

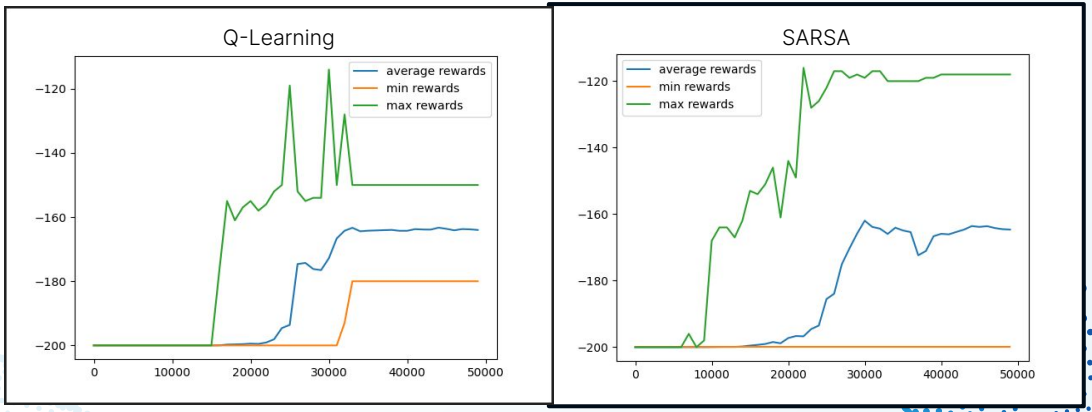
Episode Iterations = 200

Learning Rate = 0.1 * 5 = 0.5

Discount = 0.95

Epsilon = 0.5

Discrete States = 400



Mountain Car - Compare and Contrast (Discount Decrease - Factor of 3)

Hyperparameters:

Episodes = 50,000

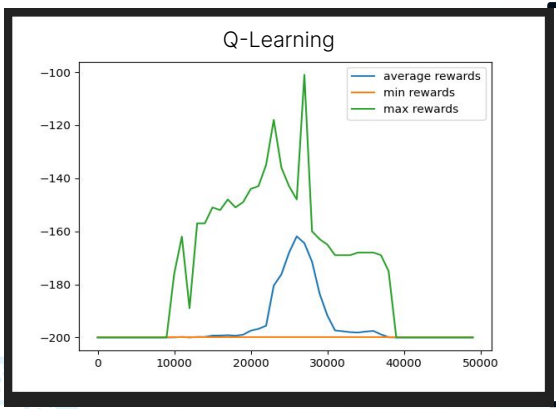
Episode Iterations = 200

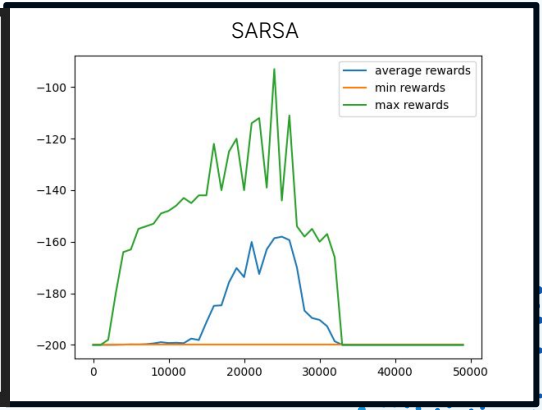
Learning Rate = 0.1

Discount = 0.95 / 3 = 0.31

Epsilon = 0.5

Discrete States = 400





Mountain Car - Compare and Contrast (Episode Iterations Decrease)

Hyperparameters:

Episodes = 50,000

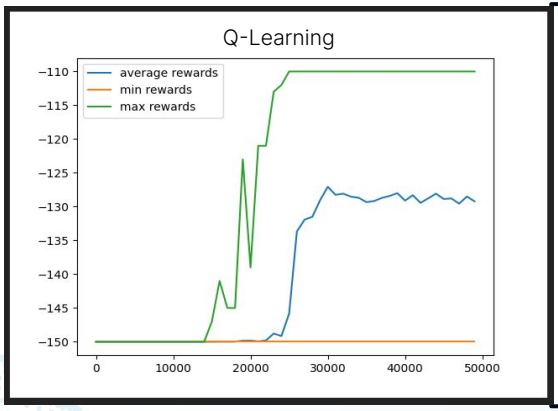
Episode Iterations = 150

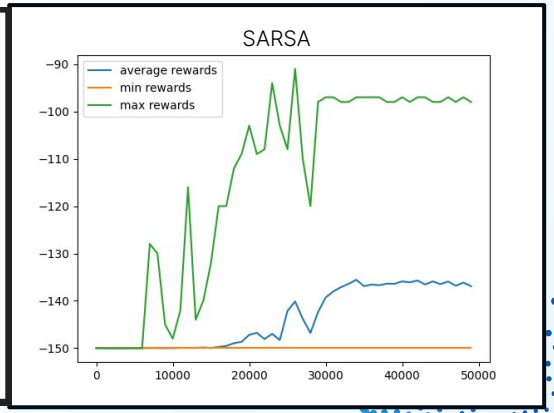
Learning Rate = 0.1

Discount = 0.95

Epsilon = 0.5

Discrete States = 400



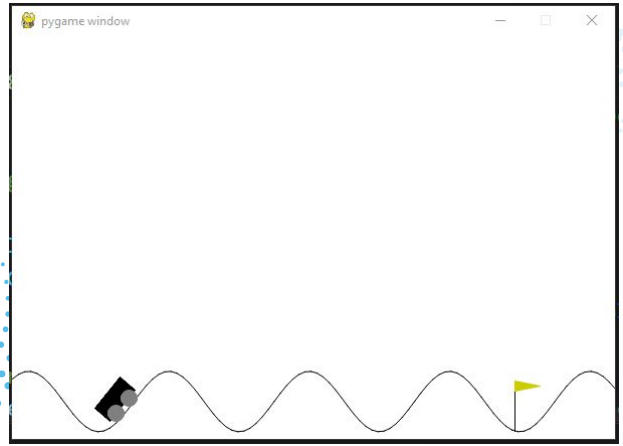


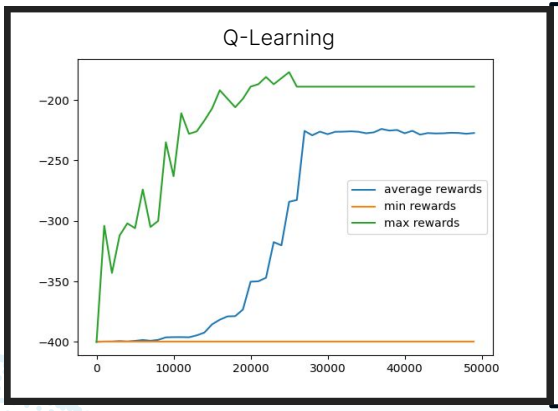
Mountain Car - Compare and Contrast (Standard - Environment Tweak)

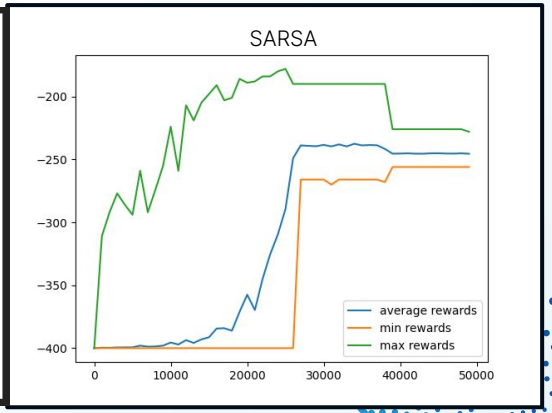
Hyperparameters:

Same as standard - With Episode Iterations = 400

Changed Environment:



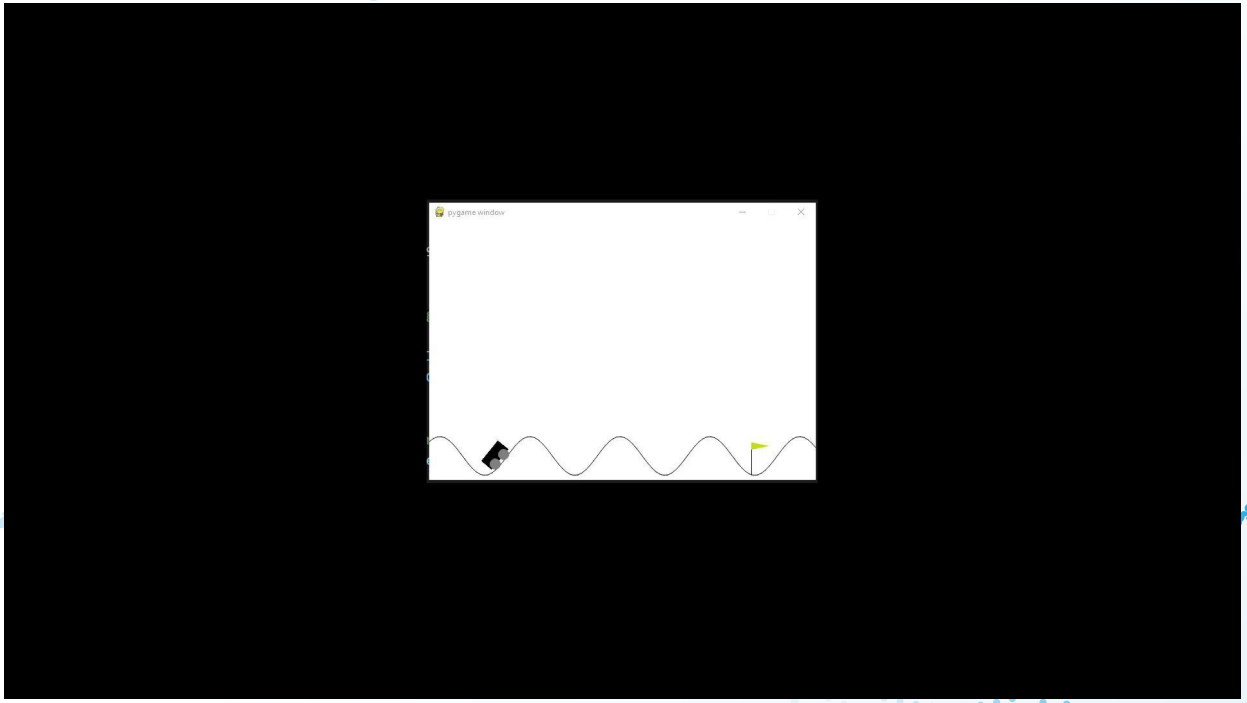




Mountain Car - Demo
Q-Learning

total reward: -115.0

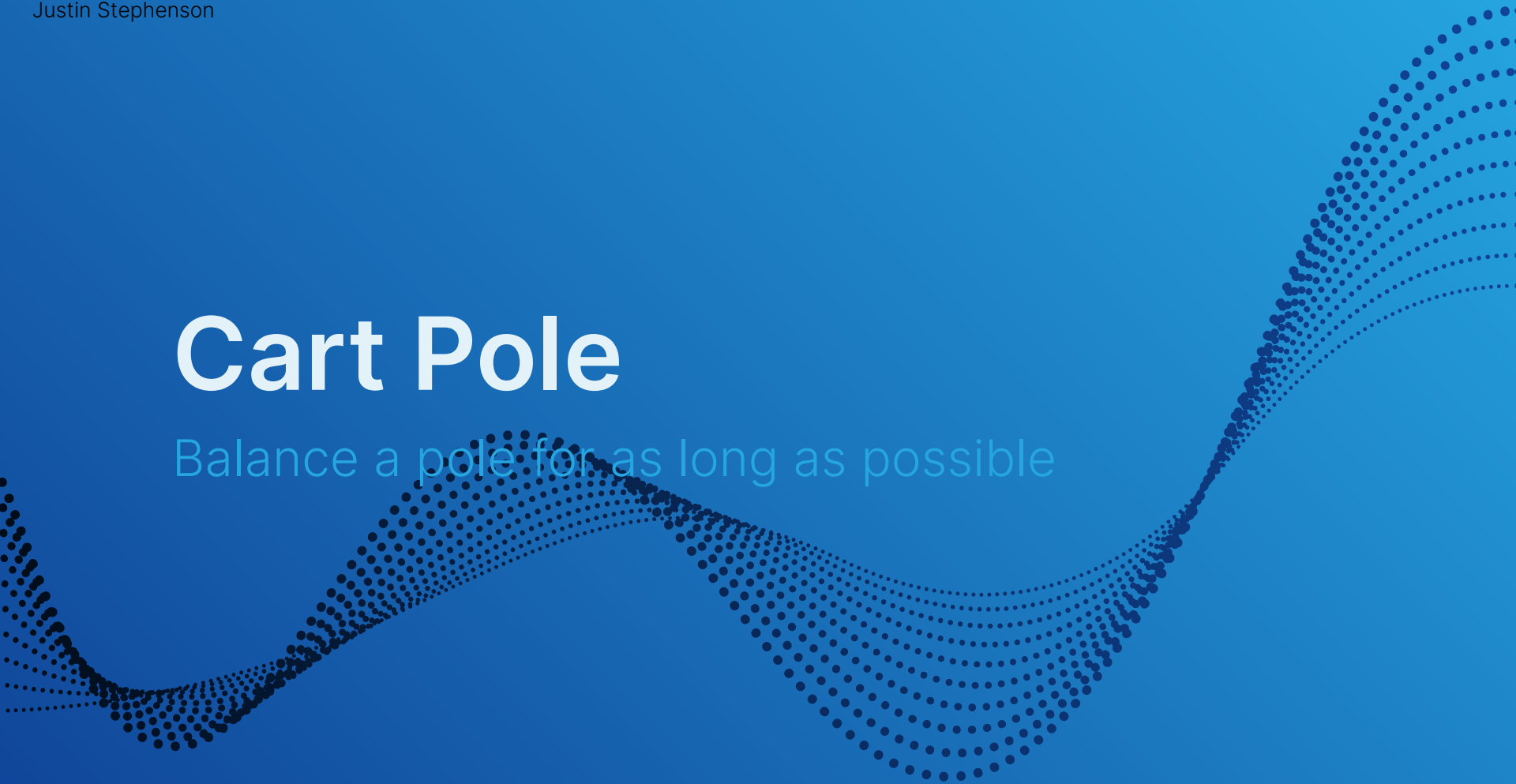
Mountain Car - Demo
Q-Learning - Environment Tweak



Justin Stephenson

Cart Pole

Balance a pole for as long as possible



Cart Pole - Description

Goal: Balance a pole on a cart for as long as possible, limit of 500 episode iterations. Plus keep the cart inside of the window boundaries.

Actions:

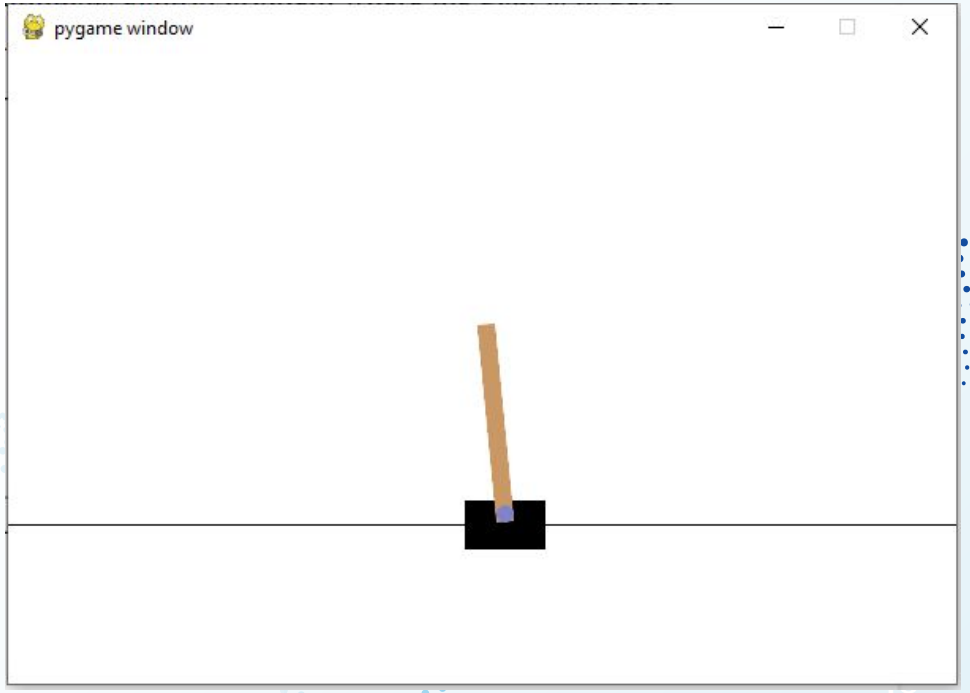
- Move cart to the left
- Move cart to the right

States: A combination of:

- Cart position on x-axis
- Cart velocity
- Pole angle
- Pole angular velocity

Rewards:

- +1 for episode iteration
- -375 for failing



Justin Stephenson

Cart Pole - Compare and Contrast (Standard)

Hyperparameters:

(Q-Learning & SARSA)

Episodes = 50,000

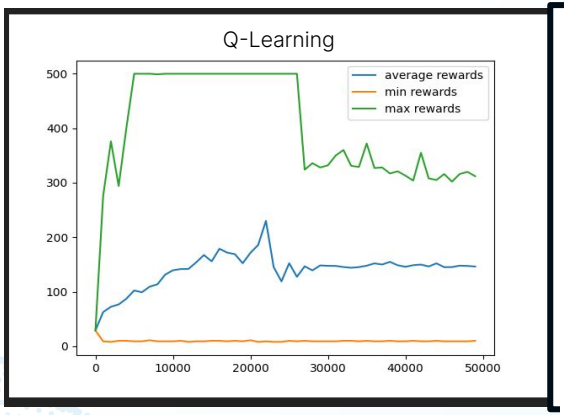
Episode Iterations = 500

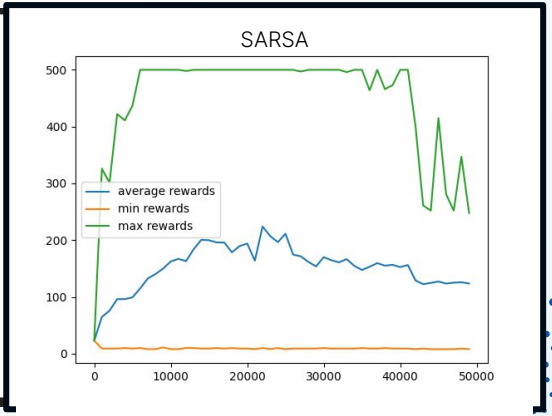
Learning Rate = 0.1

Discount = 0.95

Epsilon = 0.5

Discrete States = 160,000





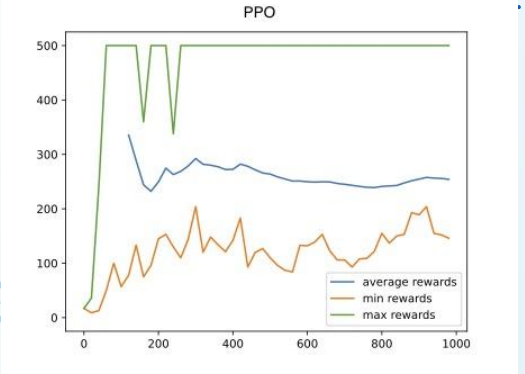
Hyperparameters:(PPO)

Episodes = 1000

Learning Rate = 0.0002

Discount = 0.99

Clip = 0.1



Cart Pole - Compare and Contrast (Learning Rate Increase - Factor of 5)

Hyperparameters:

(Q-Learning & SARSA)

Episodes = 50,000

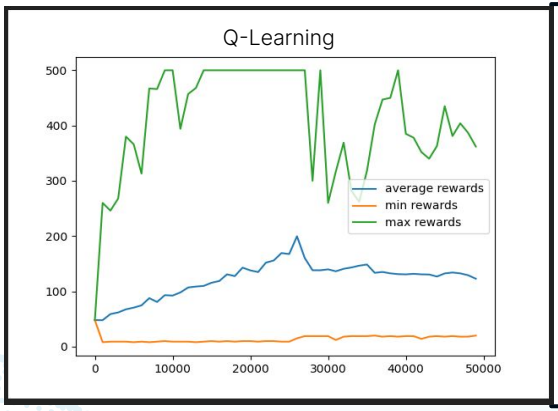
Episode Iterations = 500

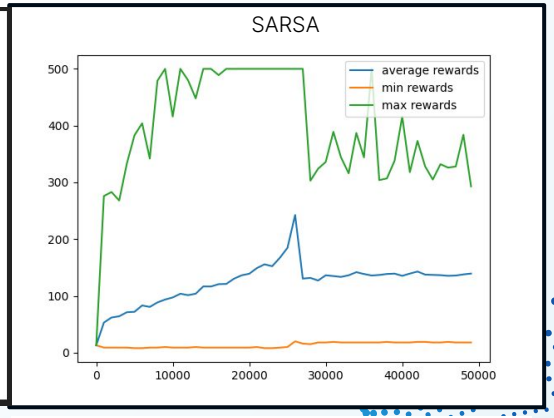
Learning Rate = 0.1 * 5 = 0.5

Discount = 0.95

Epsilon = 0.5

Discrete States = 160,000





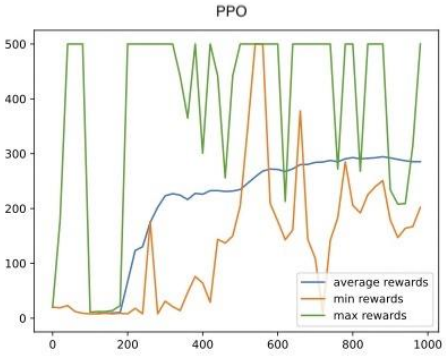
Hyperparameters (PPO)

Episodes = 1000

Learning Rate = 0.0002 * 5 = 0.001

Discount = 0.99

Clip = 0.1



Cart Pole - Compare and Contrast (Discount Decrease - Factor of 3)

Hyperparameters:

Episodes = 50,000

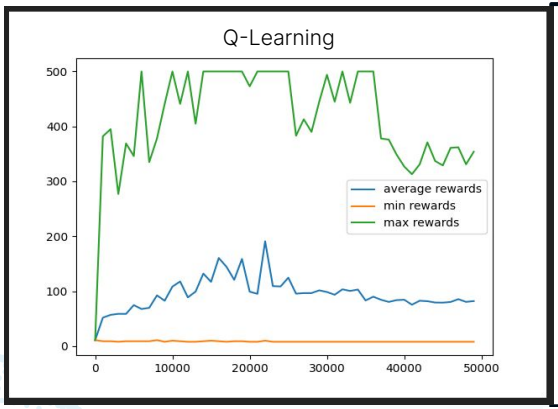
Episode Iterations = 500

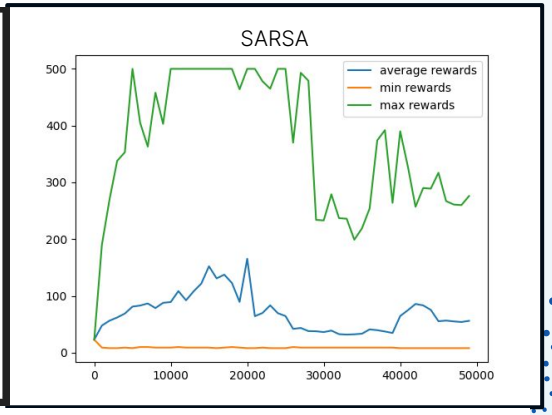
Learning Rate = 0.1

Discount = 0.95 / 3 = 0.31

Epsilon = 0.5

Discrete States = 160,000





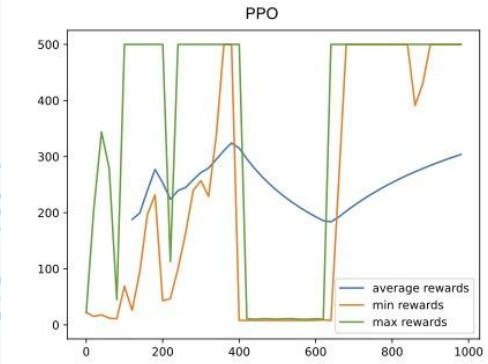
Hyperparameters:

Episodes = 1000

Learning Rate = 0.0002

Discount = 0.99 / 3 = 0.33

Clip = 0.1



Cart Pole - Compare and Contrast (Episode Iterations Increase)

Hyperparameters:

Episodes = 50,000

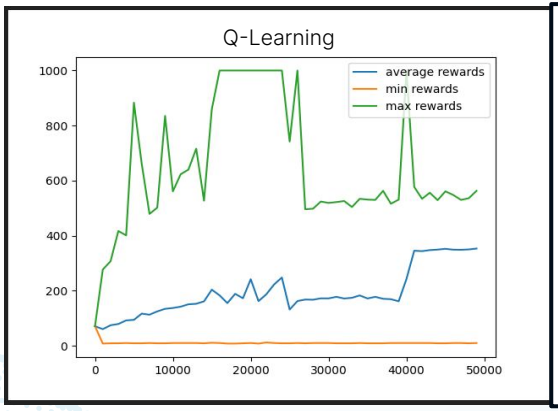
Episode Iterations = 1000

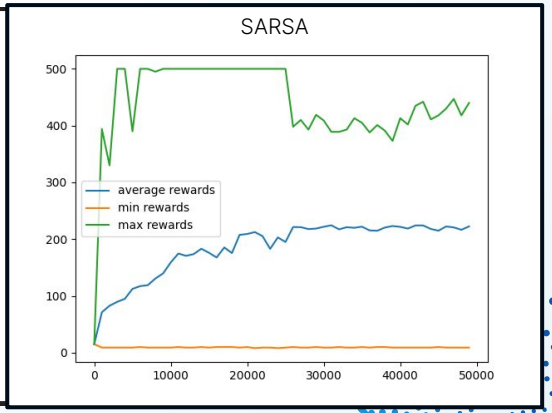
Learning Rate = 0.1

Discount = 0.95

Epsilon = 0.5

Discrete States = 160,000



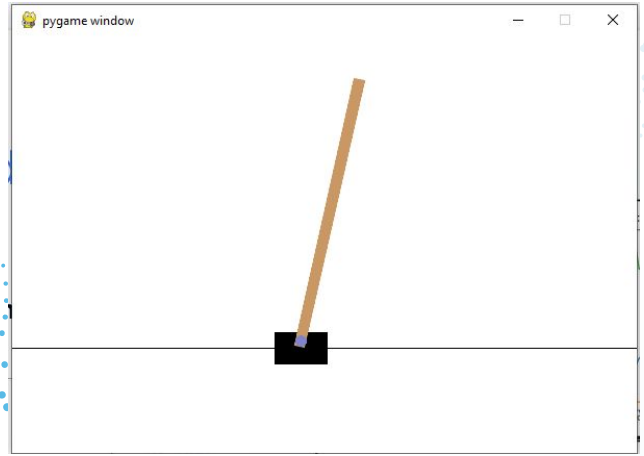


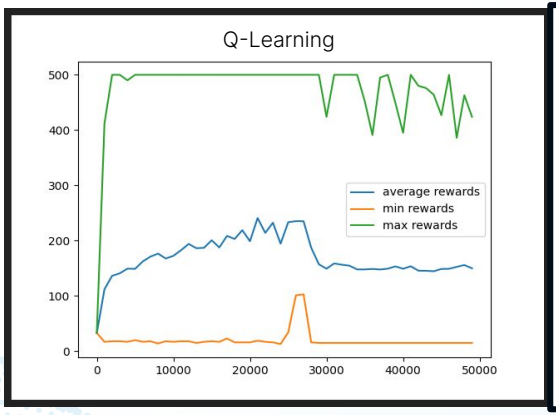
Cart Pole - Compare and Contrast (Standard - Environment Tweak)

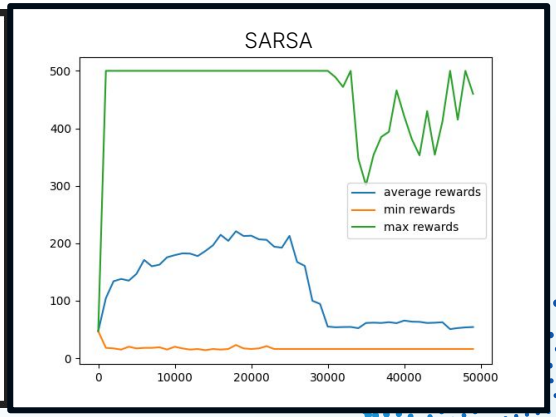
Hyperparameters:

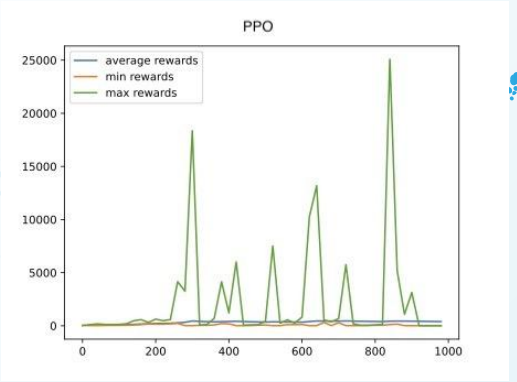
Same as standard

Changed Environment:

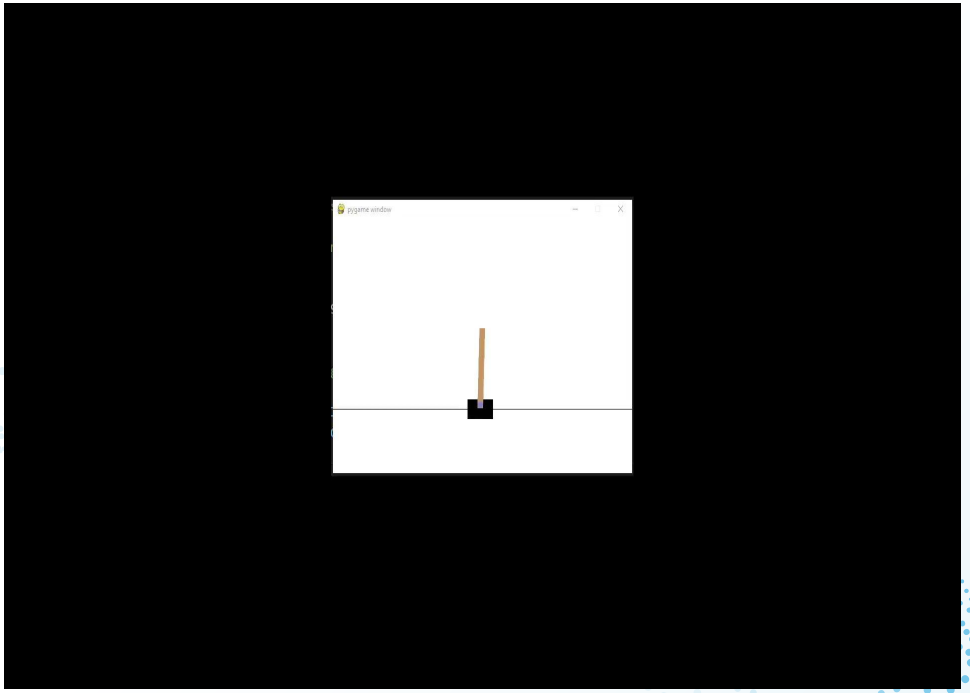








Cart Pole - Demo
Q-Learning



total reward: 217.0

Cart Pole - Demo
PPO - No time limit

total reward: 2400