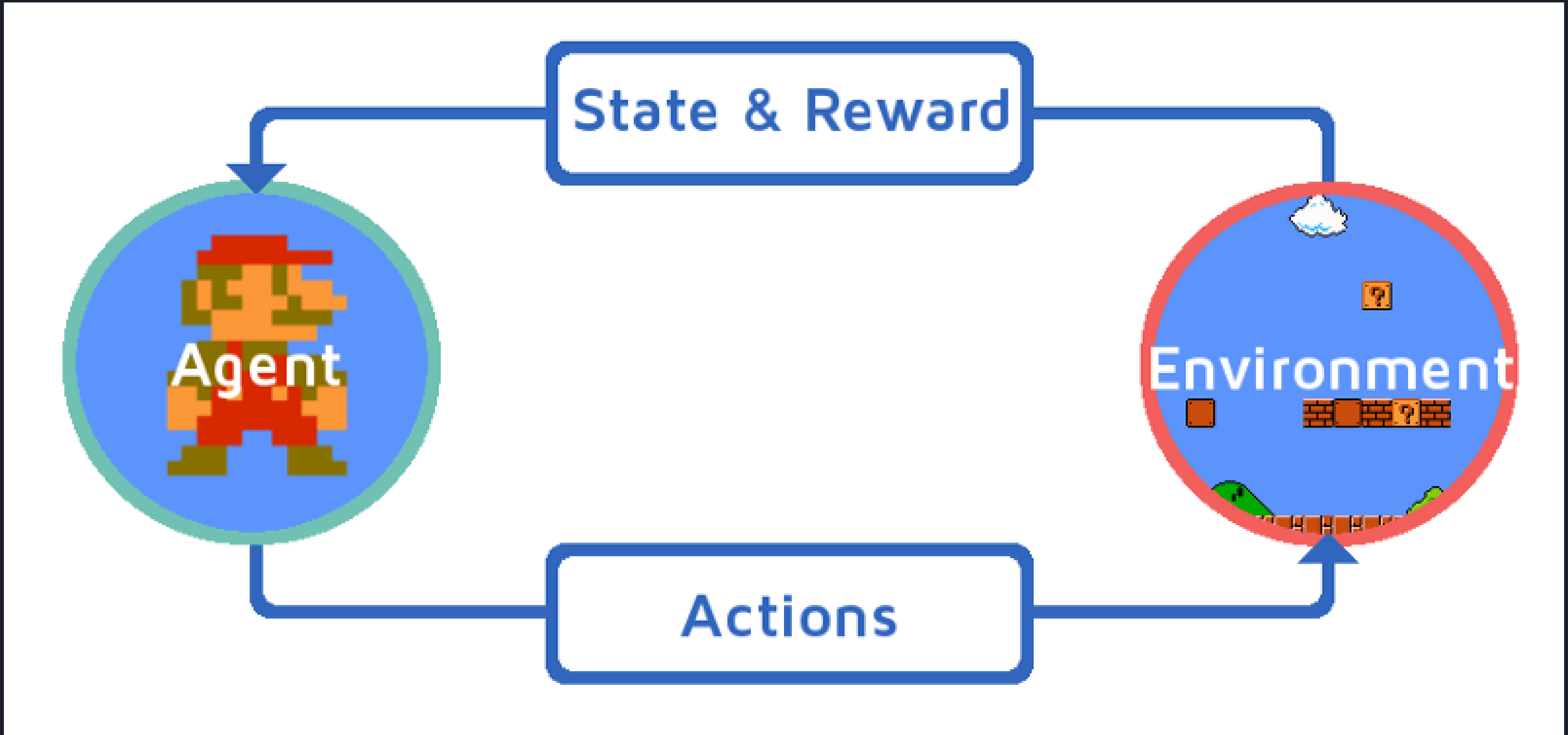
An Introduction to Reinforcement Learning

Solving the Cart Pole Problem

What is Reinforcement Learning?


State & Reward
Agent
Environment
Actions

Markov Decision Process

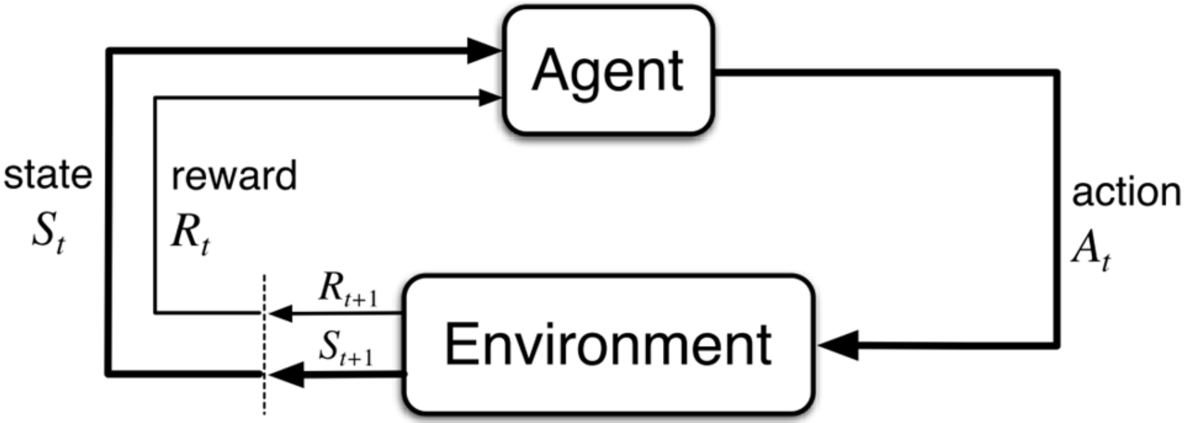


Figure 3.1: The agent–environment interaction in a Markov decision process.

$$v_\pi(s) = \sum_a \pi(a|s) \sum_{s',r} p(s', r \,|\, s, a)\Big[r + \gamma v_\pi(s')\Big], \quad \text{for all } s \in \mathcal{S},$$

The Bellman Equation

Q-Learning

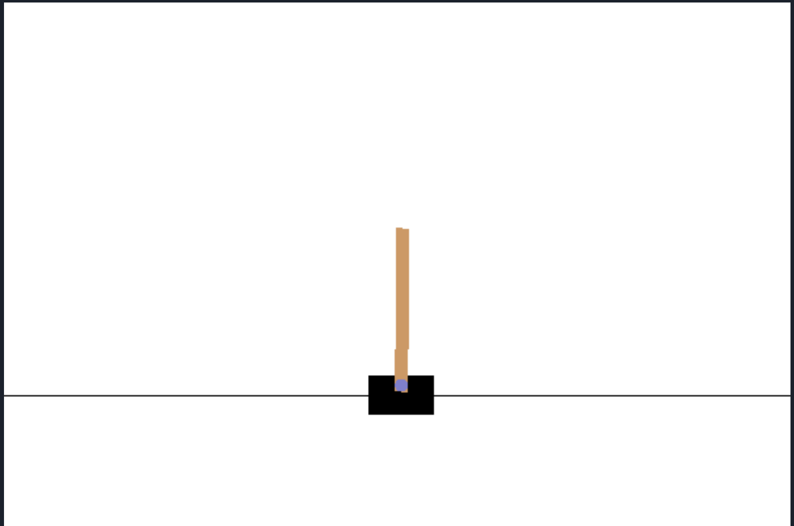
- Q-table
  - Set of state/action pairs
  - Q-values
- Iterate and update

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\left[r_{t+1} + \lambda \max_a Q(s_{t+1}, a) - Q(s_t, a_t)\right]$$

Cart Pole Problem

A pole is attached by an un-actuated joint to a cart, which moves along a frictionless track. The system is controlled by applying a force of +1 or -1 to the cart. The pendulum starts upright, and the goal is to prevent it from falling over. A reward of +1 is provided for every timestep that the pole remains upright. The episode ends when the pole is more than 15 degrees from vertical, or the cart moves more than 2.4 units from the center.

https://gym.openai.com/envs/CartPole-v1/



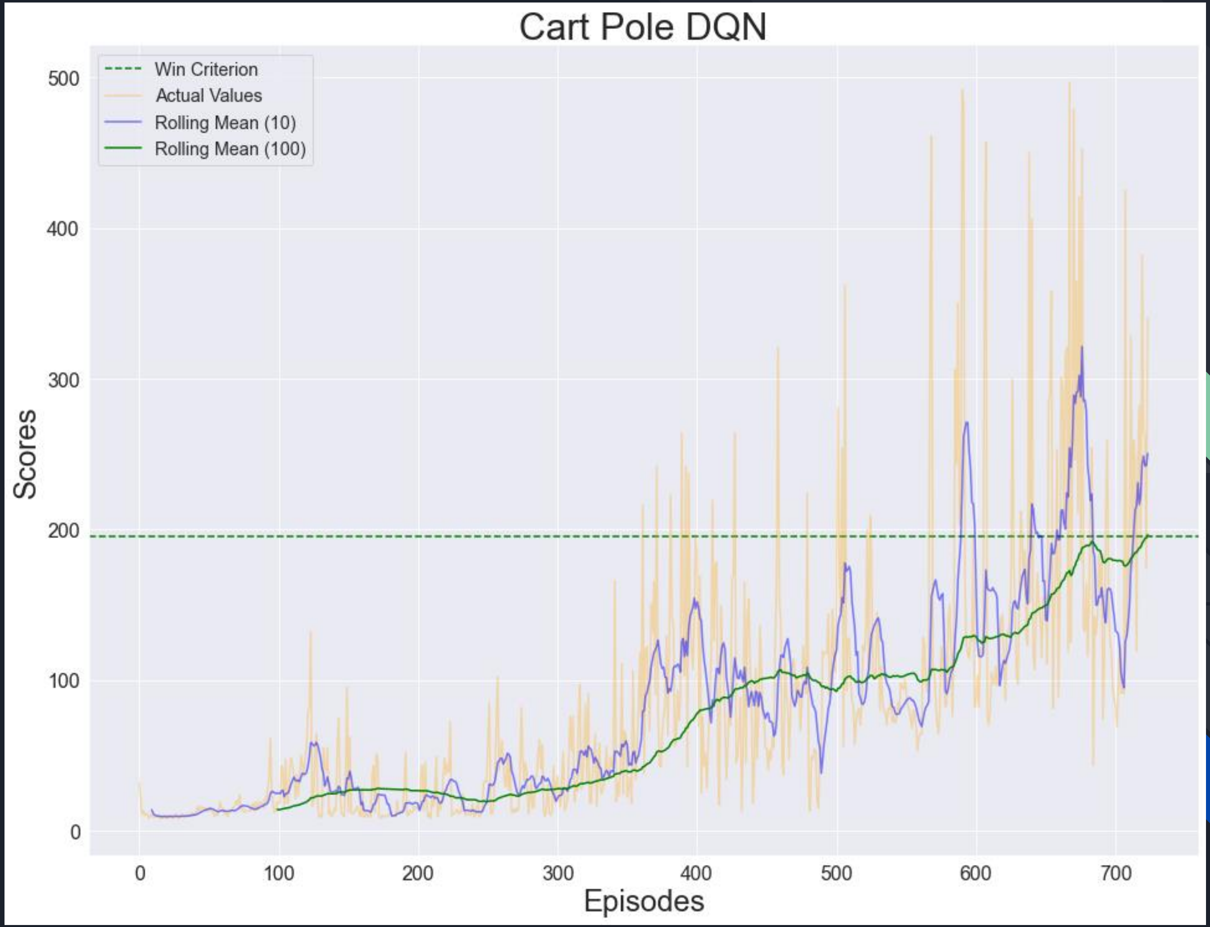
The Process

- 2 layer dense NN
  - 24 nodes each, ReLU
  - Linear output
- Adam optimizer
  - Alpha = .001
- Epsilon
  - Decayed from 1.0 to .01 at a rate of .995 per episode
- Gamma = .9

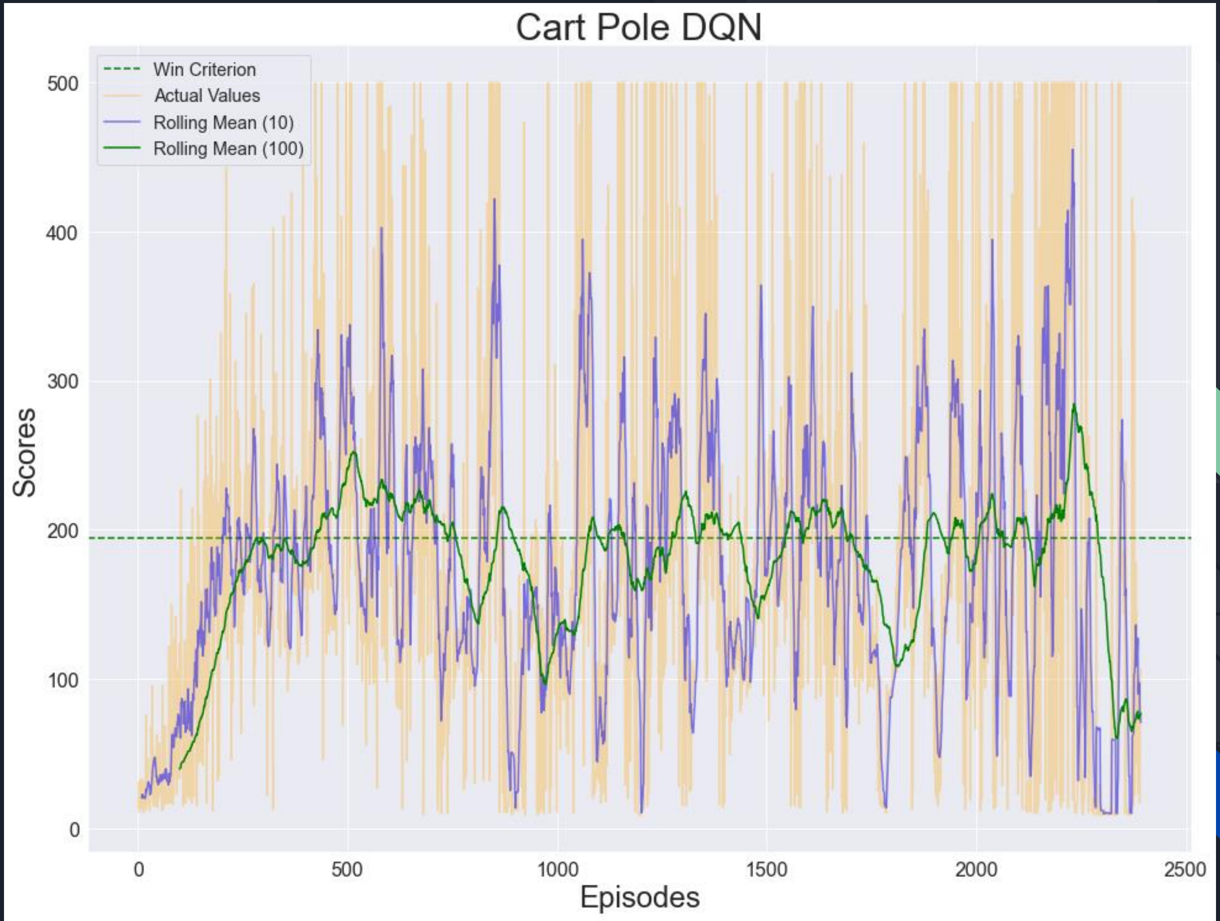
The Environment

Each state consists of:

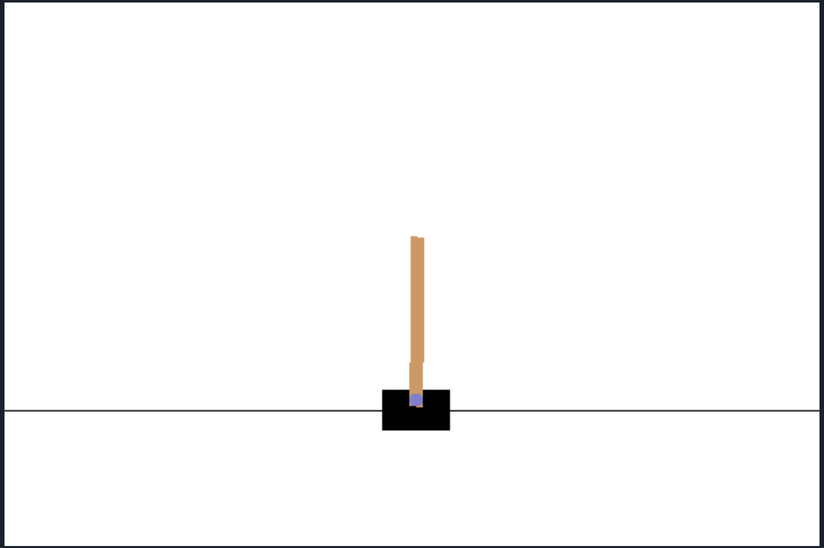
- Cart position
- Cart velocity
- Pole angle
- Pole angular velocity



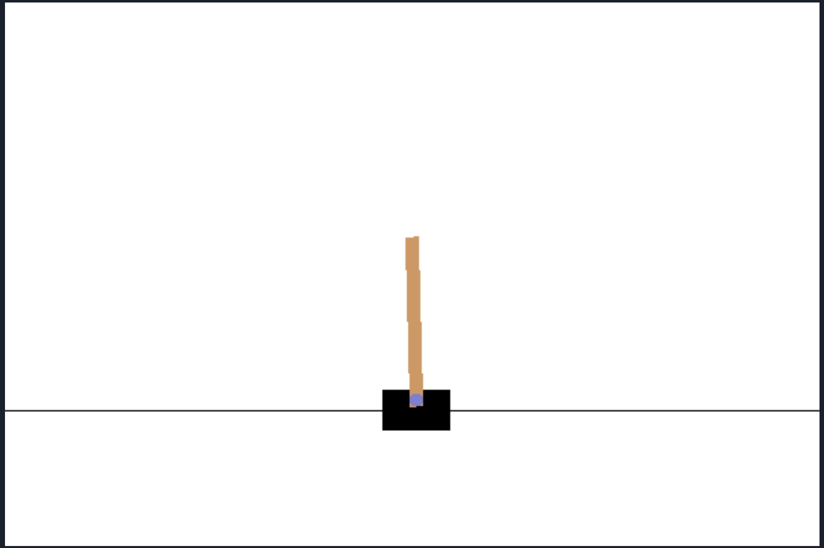
Cart Pole DQN
Win Criterion
Actual Values
Rolling Mean (10)
Rolling Mean (100)
Scores
Episodes
0
100
200
300
400
500
600
700


Cart Pole DQN
Win Criterion
Actual Values
Rolling Mean (10)
Rolling Mean (100)
Scores
0
100
200
300
400
500
Episodes
0
500
1000
1500
2000
2500

Before



After



Applications

- Fanuc Robotics
- Tesla and other EV companies
- Google data center cooling
- Stock trading
- Supply chain logistics

References

- Reinforcement Learning: An Introduction by Sutton and Barto
- Introduction to RL by Greg Surma
- Wikipedia:
  - Reinforcement Learning
  - Markov Decision Process
- Randomant.com on Q-Learning