

# DSAI - HW4: Classic control in reinforcement learning

## Mountain Car v0

The Project used Q-Learning

How to implement Radial Basis Function(RBF)?

- RBFsampler is inside **Sci-Kit Learn**
- RBFsampler uses a **Monte Carlo algorithm**
- The game don't need epsilon-greedy
- 1-hidden layer neural network, with RBF kernel as activation function
- Used to convert a state to a featurized representation.
- Used RBF kernels with different variances to cover different parts of the space
- Exemplars in `env.observation_space.sample()` in the project used **500 exemplars**
- Info: <https://github.com/dennybritz/reinforcement-learning>

## Model Class

### Initialize

- Holds one SGDRegressor for each action
- Assigned instance variables and instantiate our SGDRegressor

### Predict

- Transforms the state into a feature vector and makes a prediction of values one for each action

### Update

- Transforms the input state into a feature vector

### Sample Action

- Performs the Epsilon greedy

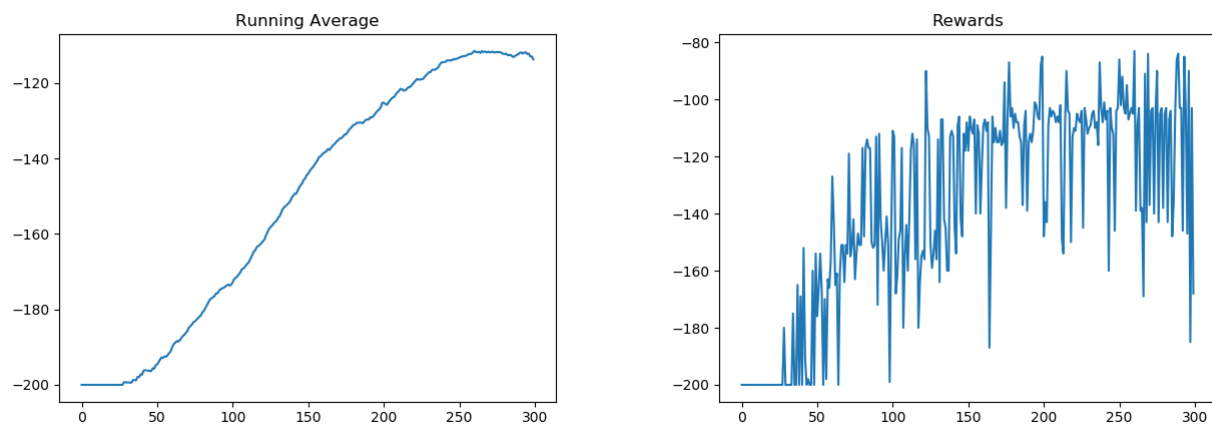
## Questions:

What kind of RL algorithms did you use? value-based, policy-based, model-based? why? (10%)

- Value-Based : Q-Learning
- I selected Q-Learning because it determines which action is best in the current state as well as all future states. It is an algorithm that attempts that takes an observer as input and returns an action as output. Which is what I need to implement in the Mountain Car.

This algorithms is off-policy or on-policy? why? (10%)

- On-Policy, because our agent is trying to learn the job.
- Base on the graph the agent starts with a -200 reward point but after around 50 episodes, the agent started to understand how to play the game. The agent is using gamma of 0.99 so it won't be seeking for reward everywhere. After all the agent under stand which action to use to get better rewards.



How does your algorithm solve the correlation problem in the same MDP? (10%)

- When playing the same game, maybe implementing a single policy network to reuse the action will be great. Actor-Mimic is a solution, because it use a the best DQN as basis to train our model. I've seen <https://github.com/eparisotto/ActorMimic> project and the result were nice.