CSC 449: Programming Assignment 3

Christian Olson

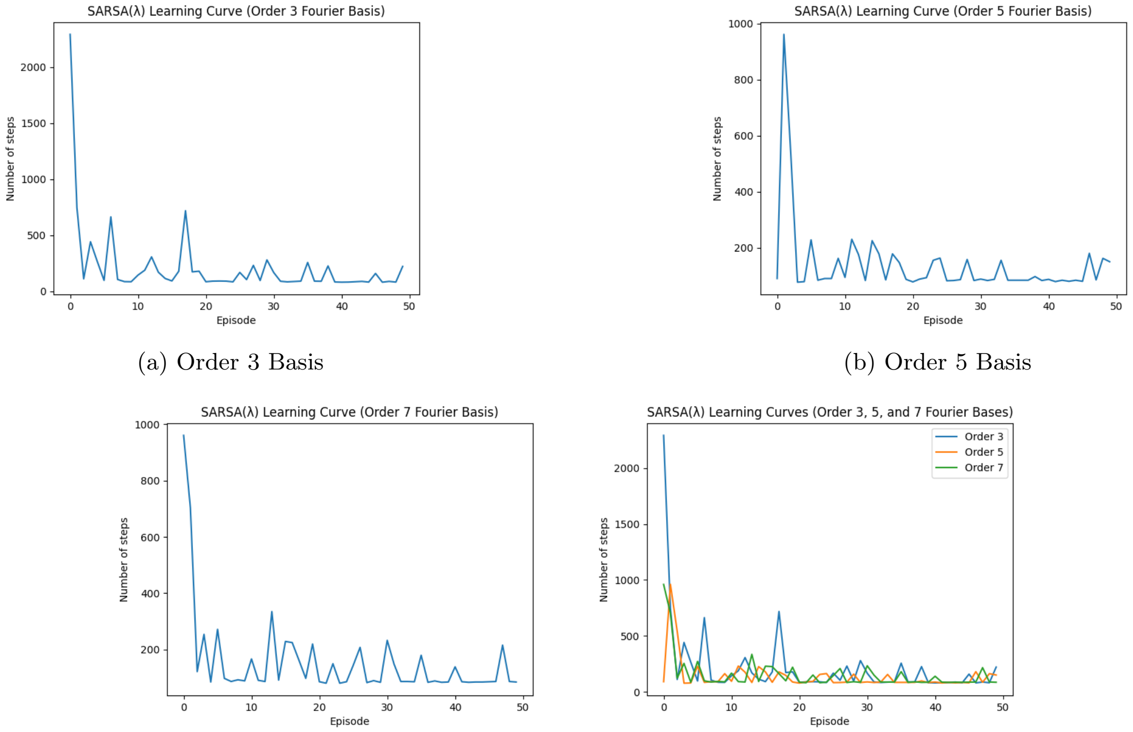
December 6, 2022

1 Overview

This assignment describes an implementation of SARSA(λ) which learns to solve the Mountain Car problem. The implemented SARSA(λ) algorithm is a True Online SARSA(λ) algorithm which uses a Fourier Basis. The Fourier basis for this algorithm is of order 3, 5, or 7 over 3 dimensions. These 3 dimensions are the two state dimensions, position and velocity, and the action space. The algorithm is also designed to be ϵ-greedy when choosing actions using $q(s, a, \mathbf{w})$. Over 1000 episodes, this algorithm is run with the values $\alpha = 0.001$, $\epsilon = 0$, $\gamma = 1$, and $\lambda = 0.9$.

The associated code files can be complied via `main.py` with the command: `python main.py` This program runs the SARSA(λ) algorithm with order 3, 5, and 7 Fourier basis and displays the results similar to the section below.

2 Results



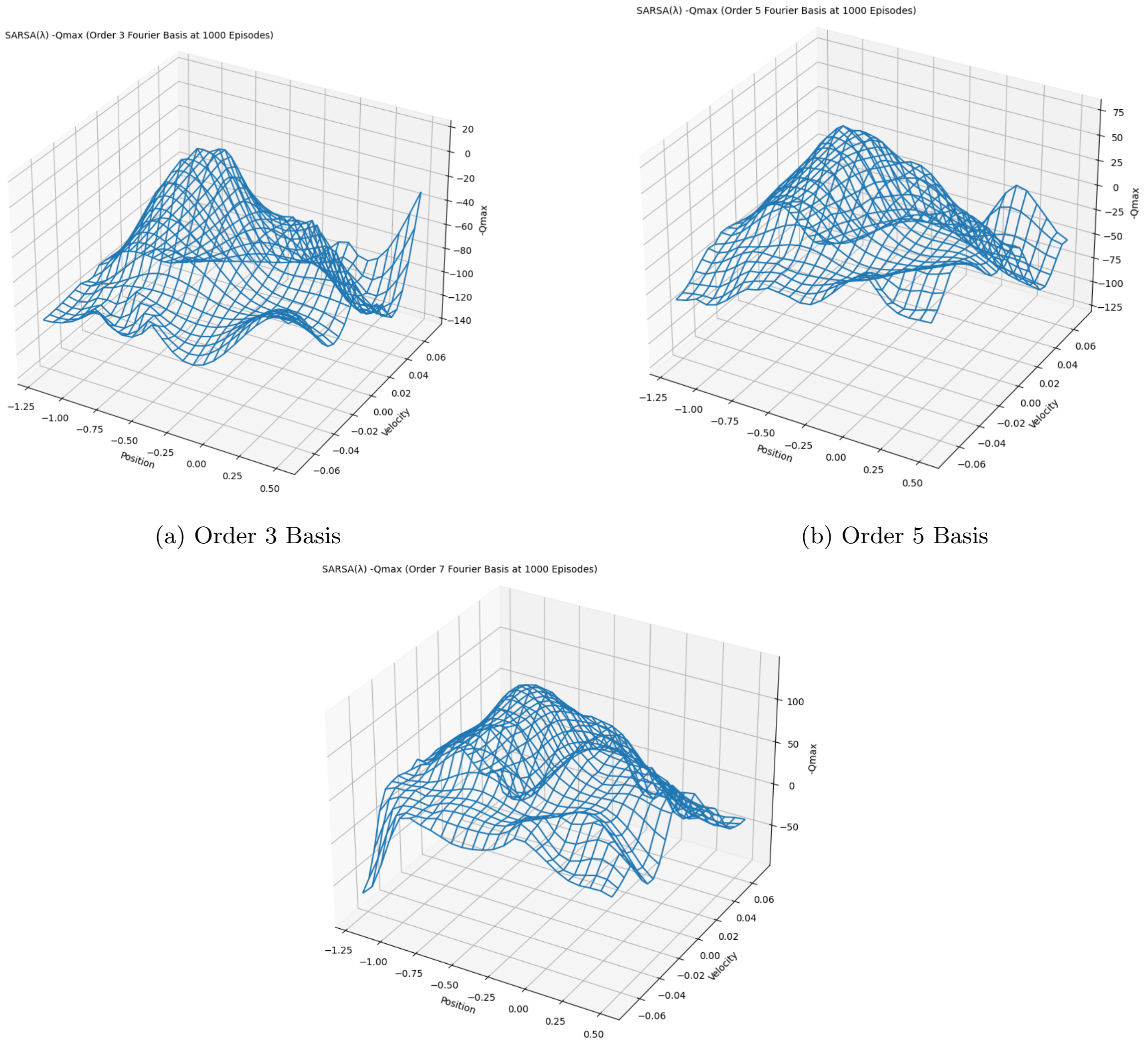
(a) Order 3 Basis

(b) Order 5 Basis

(c) Order 7 Basis

(d) All Bases

Figure 1: Learning Curves for SARSA(λ) with Fourier Bases of order 3, 5, 7 over 50 episodes. The curves show that increase in order somewhat affects the rate at which the algorithm can learn as higher orders require fewer steps to per episode, but more computation time as they compute more features.



(a) Order 3 Basis

(b) Order 5 Basis

(c) Order 7 Basis

Figure 2: Surface plots of the $-\max aq(s, a, \mathbf{w})$ learned by the algorithm with different order bases. The lower order surfaces are much smoother than those of higher order as they do not have the precision required to accurately compute certain local extrema.

3 Discount Rate γ

Given the current negative step reward and zero goal reward, $\gamma < 1$ would lessen the impact of negative reward from previous steps and make action more independent of one another. In the mountain car problem, acceleration of the car is a significant factor in its ability to learn. A lower γ value would reduce the impact of acceleration on the problem and significantly increase the difficulty of learning.

Given a zero step reward and positive goal reward, circumstances would be learning would behave very differently. At $\gamma = 1$, the algorithm would struggle to find the goal during the first few episodes as it is given no guidance by the zero step reward. If it was able to reach the goal repeatedly, the reward would slowly percolate down from this point. This is not desirable as it teaches the car what to do only near the goal and not how to get there. If $\gamma < 1$ the reward would spread just as quickly as with $\gamma = 1$, but with less strength.