

Homework 4

*Handed Out: March 26, 2018**Due: April 9, 2018*

1. Implement the Monte Carlo ES (Exploring Starts) algorithm (see p.81 in textbook). Complete *MCES.py*. Run *blackjack_control.py* to test your code on the blackjack example (see Example 5.3). Your results should be similar to Figure 5.2 (the error rate should be around 5% for 100,000 episodes). Submit *MCES.py* and the two figures you generated on collab. (20 points)
2. Implement Sarsa. Complete *sarsa.py*. Then run *windy.py* to test your code on the windy gridworld example (Example 6.5). You should get a figure (episodes vs time steps) similar to Figure 6.3. To get the optimal route, you may need to run *windy.py* a few times. Append the optimal route as comments to *sarsa.py*. Submit *sarsa.py* and the figure you generated. (20 points)
3. (based on Exercise 6.9) Re-solve the windy gridworld task with 9 possible actions: left (action 0), up (1), right (2), down (3), up/left (4), up/right (5), down/right (6), down/left (7), stand(8). Specifically, complete *windy9_setup.py*. Run *windy9.py*. Append the optimal route (you may need to run the file several times) as comments to *windy9_setup.py*. Submit *windy9_setup.py* and the produced figure. (20 points)
4. Implement Sarsa(λ). Complete *sarsa_lambda.py*. Run *windy_sarsaLambda.py* to test your code on the windy gridworld example. Does λ have any impact on the learning curve? Submit *sarsa_lambda.py* and the figure you generated. (20 points)