

CS747 : Foundations of Intelligent & Learning Agents

Programming Assignment 3

REPORT

Vinit P. Doke
190260018¹

¹ Email: 190260018@iitb.ac.in

November 2, 2021

Content

1 Task 1 : SARSA Tabular	1
1.1 State Discretization and Representation	1
1.2 Other Hyper-parameters	1
1.3 Plots & Results	1
2 Task 2 : SARSA with Linear Function Approximation	2
2.1 Discretization & Tile Coding	2
2.2 Other Hyper-parameters	2
2.3 Plots & Results	2
3 Screenshot of verifyOutput.sh	3
References	3

1 Task 1 : SARSA Tabular

1.1 State Discretization and Representation

The position space varies from $[-1.2, 0.6]$ and is divided into **25 uniform segments**.

The velocity space varies from $[-0.07, 0.07]$ and is divided into **25 uniform segments** too.

The index of the final state is then calculated as the follows:

The bin numbers of the (x, v) observation are (x', v') and the index is then

$$i = x' + (v' - 1) \times 25$$

The corresponding weight is stored in a 625×3 Matrix : $25 \times 25 \times 3$ (*Actions*)

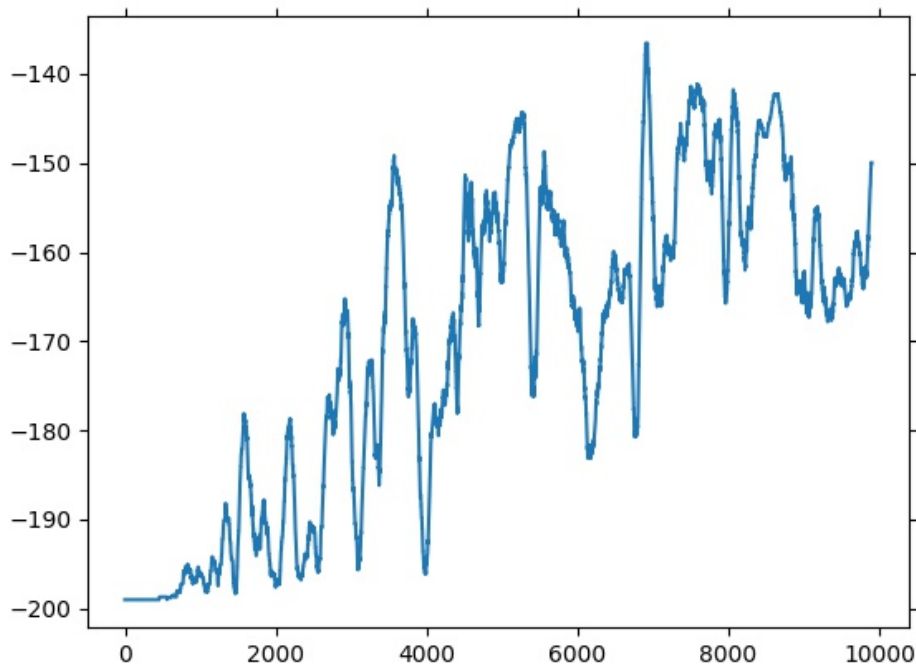
1.2 Other Hyper-parameters

- Learning Rate $\alpha = 0.1$
- Discount Factor $\gamma = 0.9$
- Epsilon $\epsilon = 0.001$

1.3 Plots & Results

Average Reward for 100 Episodes is :

-144.78



(a)

Figure 1: On X - Axis : Number of Episodes
On Y - Axis : Reward

2 Task 2 : SARSA with Linear Function Approximation

2.1 Discretization & Tile Coding

Number of Tilings Used : 12

Divisions per Tile per Dimension (both position and velocity) : 25

Position Offset : 0.0065

Velocity Offset : 0.0005

3D Weight Matrix of dimensions : $[25 \times 25, 12, 3]$

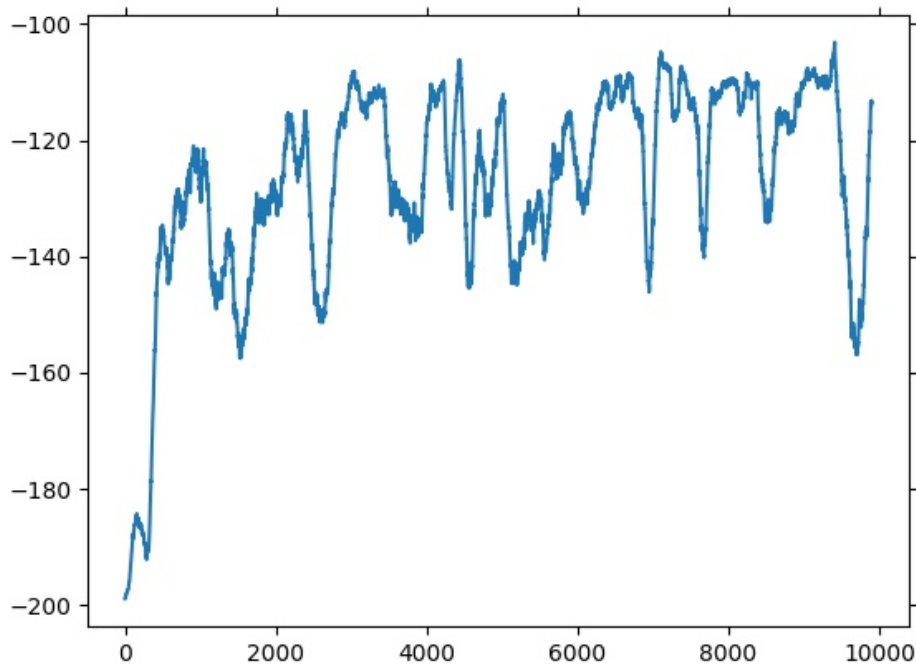
2.2 Other Hyper-parameters

- Learning Rate $\alpha = 0.1$
- Discount Factor $\gamma = 0.9$
- Epsilon $\epsilon = 0.001$

2.3 Plots & Results

Average Reward for 100 Episodes is :

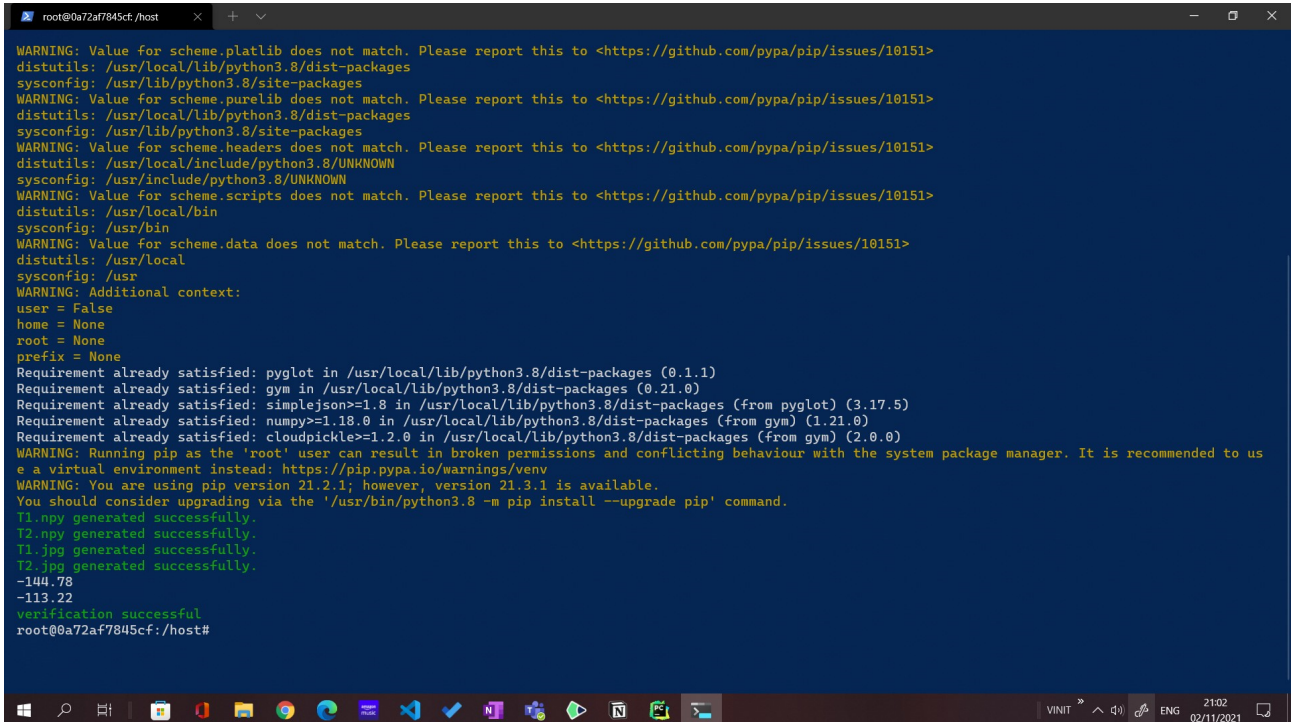
-113.22



(a)

Figure 2: Task 2 : SARSA with Tile Coding
On X - Axis : Number of Episodes
On Y - Axis : Reward

3 Screenshot of verifyOutput.sh



```

root@0a72af7845cf:/host
WARNING: Value for scheme.platlib does not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /usr/local/lib/python3.8/dist-packages
sysconfig: /usr/lib/python3.8/site-packages
WARNING: Value for scheme.purelib does not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /usr/local/lib/python3.8/dist-packages
sysconfig: /usr/lib/python3.8/site-packages
WARNING: Value for scheme.headers does not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /usr/local/include/python3.8/UNKNOWN
sysconfig: /usr/include/python3.8/UNKNOWN
WARNING: Value for scheme.scripts does not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /usr/local/bin
sysconfig: /usr/bin
WARNING: Value for scheme.data does not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /usr/local
sysconfig: /usr
WARNING: Additional context:
user = False
home = None
root = None
prefix = None
Requirement already satisfied: pyplot in /usr/local/lib/python3.8/dist-packages (0.1.1)
Requirement already satisfied: gym in /usr/local/lib/python3.8/dist-packages (0.21.0)
Requirement already satisfied: simplejson>=1.8 in /usr/local/lib/python3.8/dist-packages (from pyplot) (3.17.5)
Requirement already satisfied: numpy>=1.18.0 in /usr/local/lib/python3.8/dist-packages (from gym) (1.21.0)
Requirement already satisfied: cloudpickle>=1.2.0 in /usr/local/lib/python3.8/dist-packages (from gym) (2.0.0)
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to us
e a virtual environment instead: https://pip.pypa.io/warnings/venv
WARNING: You are using pip version 21.2.1; however, version 21.3.1 is available.
You should consider upgrading via the '/usr/bin/python3.8 -m pip install --upgrade pip' command.
T1.npy generated successfully.
T2.npy generated successfully.
T1.jpg generated successfully.
T2.jpg generated successfully.
-144.78
-113.22
verification successful
root@0a72af7845cf:/host#

```

(a)

References

1. Tile Coding
<https://towardsdatascience.com/reinforcement-learning-tile-coding-implementation-7974b600762b>
2. SARSA with LFA
<https://artint.info/2e/html/ArtInt2e.Ch12.S9.SS1.html>
3. Book : Sutton & Barto
4. OpenAI Gym
<https://gym.openai.com/docs/>