CS6700: Reinforcement Learning

Programming Assignment 2

Shubham Patel, ME19B170
Sharuhasan, EE19B117

1 Introduction

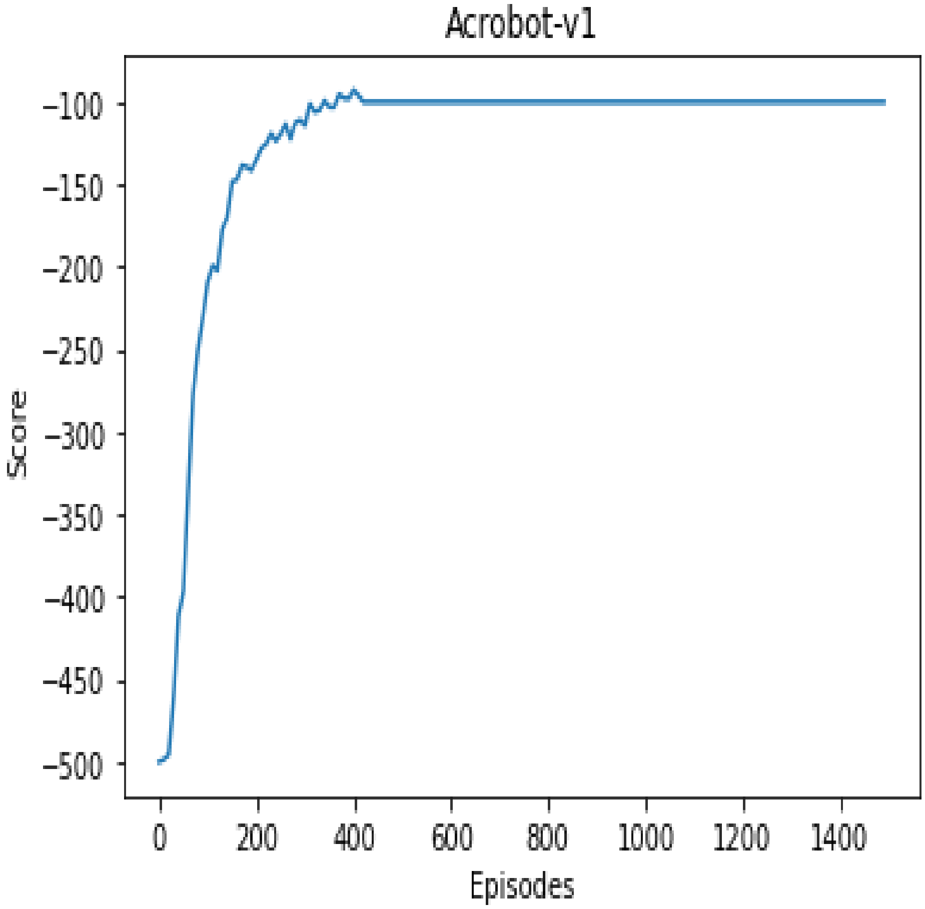
In this assignment, we implement two algorithms: **DQN** and **Actor-Critic**. There are **three** environments: `Acrobot-v1`, `CartPole-v1` and `MountainCar-v0`. For each algorithm, there are **12** configurations based on different combinations of hyperparameter values. For **DQN**, for each configuration, we present plots for rewards and steps vs episdoes. While for **Actor-Critic**, for each configuration we present 3 sub-cases (One-step, Full returns, n-step returns) each with plots of rewards and steps vs episodes.Each experiment is run for 10 times to account for stochasticity. Simulations were generated as well to better understand the behaviour of agents. Inferences for the choice of hyperparameters is also included.

2 DQN

Acrobot

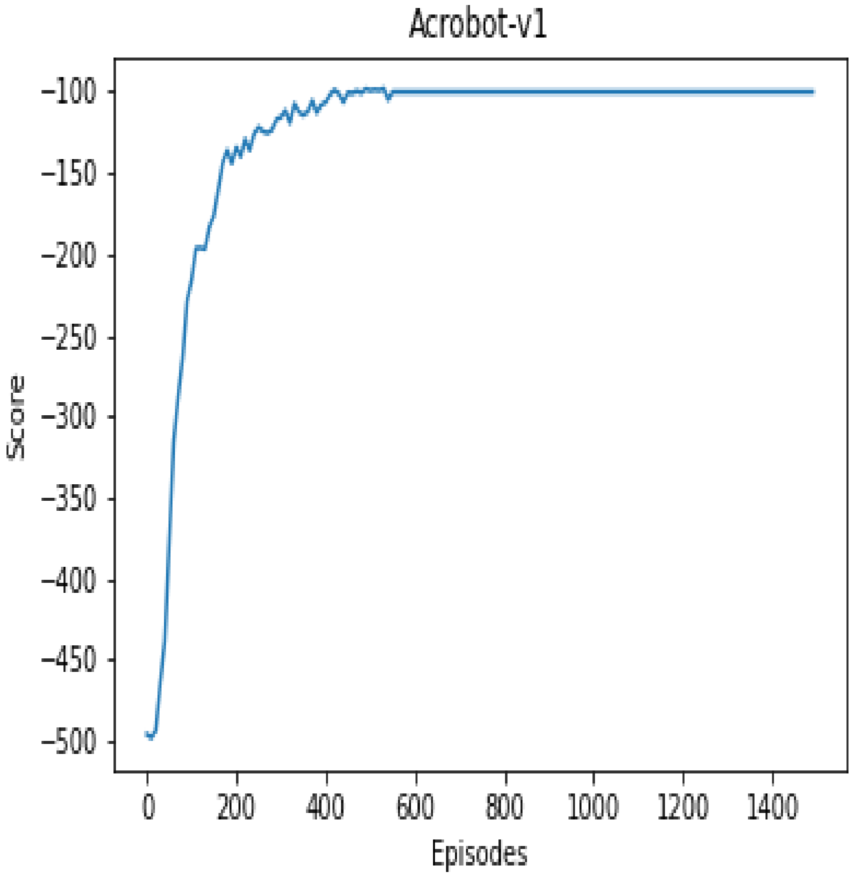
- **Hyper-parameters**
  - Learning Rate: 5e-4
  - Update Frequency: 20
  - Batch Size: 64
  - Buffer Size: 1e5
  - Architecture: 128 - 64

Results:
Average Episodes to converge = 300.6, Convergence Rate = 100%



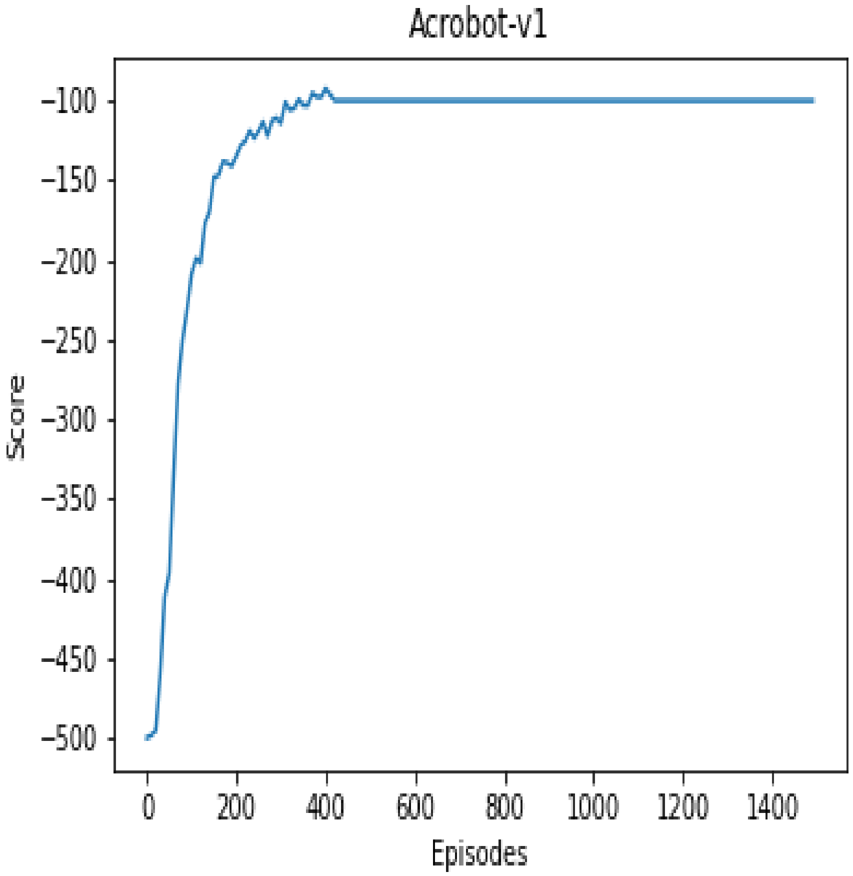
- **Hyper-parameters**
  - Learning Rate: 1e-3
  - Update Frequency: 20
  - Batch Size: 64
  - Buffer Size: 1e5
  - Architecture: 128 - 64

Results:
Average Episodes to converge = 350.2, Convergence Rate = 100%



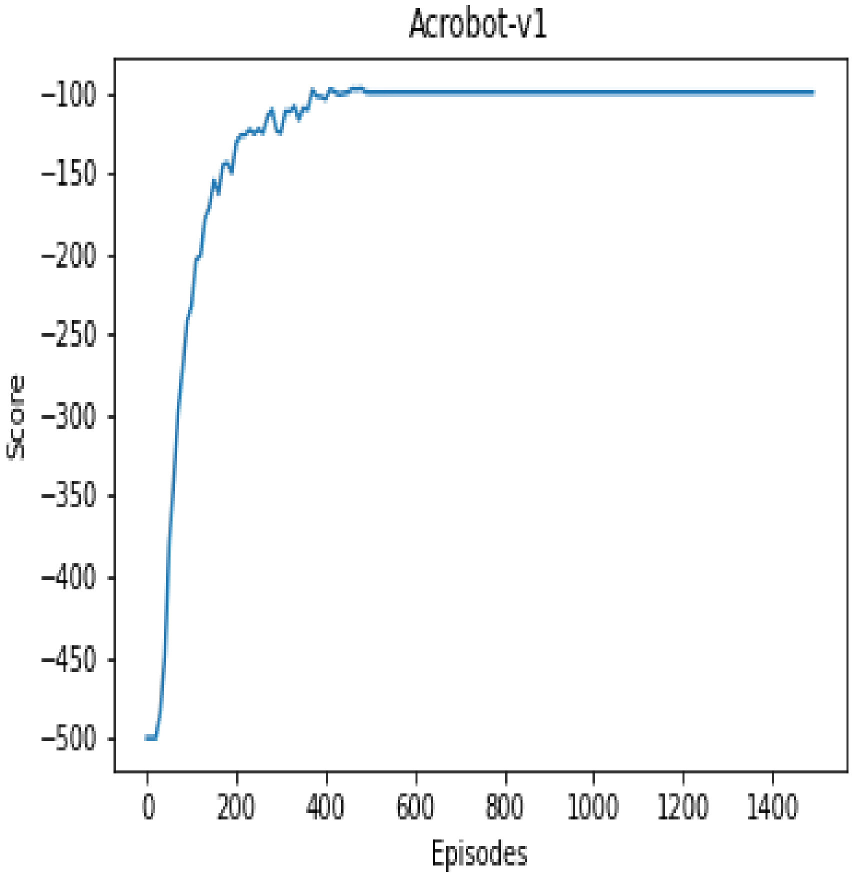
- **Hyper-parameters**
  - Learning Rate: 5e-4
  - Update Frequency: 25
  - Batch Size: 128
  - Buffer Size: 1e5
  - Architecture: 128 - 64

Results:
Average Episodes to converge = 300.6, Convergence Rate = 100%



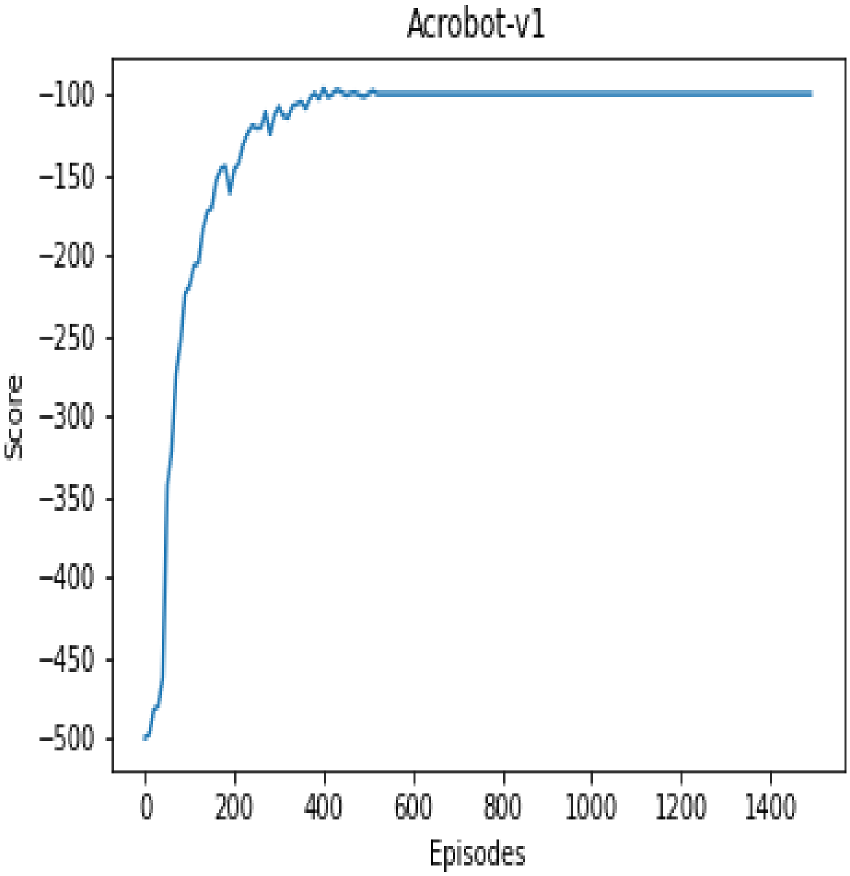
- **Hyper-parameters**
  - Learning Rate: 5e-4
  - Update Frequency: 50
  - Batch Size: 64
  - Buffer Size: 1e3
  - Architecture: 128 - 64

Results:
Average Episodes to converge = 343.8, Convergence Rate = 100%



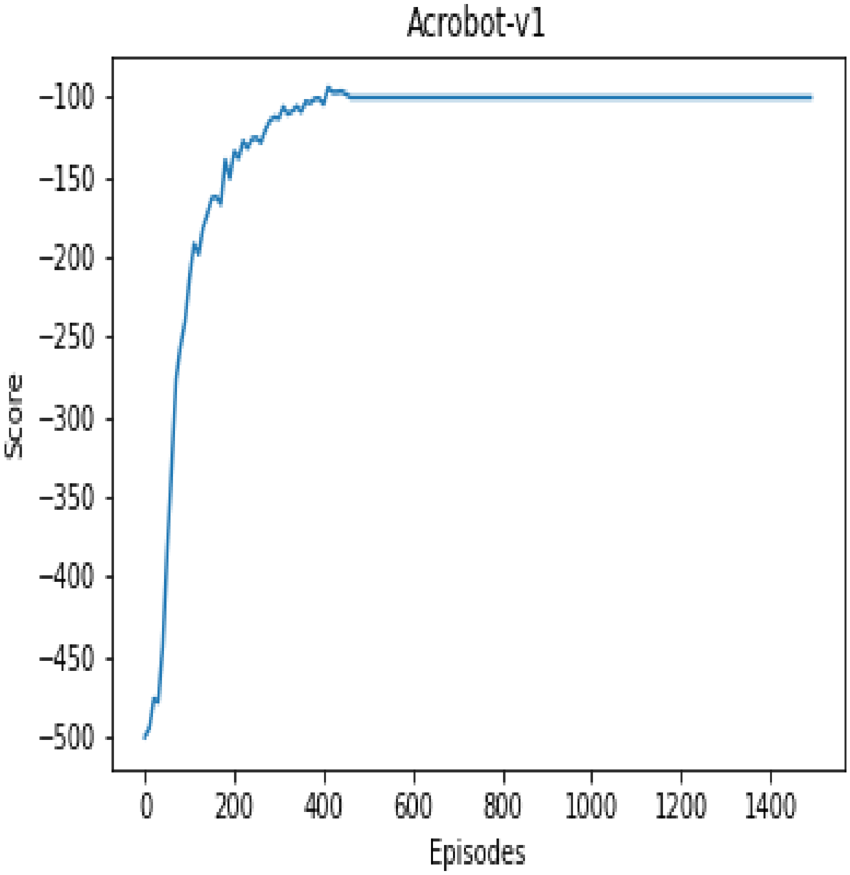
- **Hyper-parameters**
  - Learning Rate: 5e-4
  - Update Frequency: 20
  - Batch Size: 32
  - Buffer Size: 1e7
  - Architecture: 64 - 64

Results:
Average Episodes to converge = 350.6, Convergence Rate = 100%



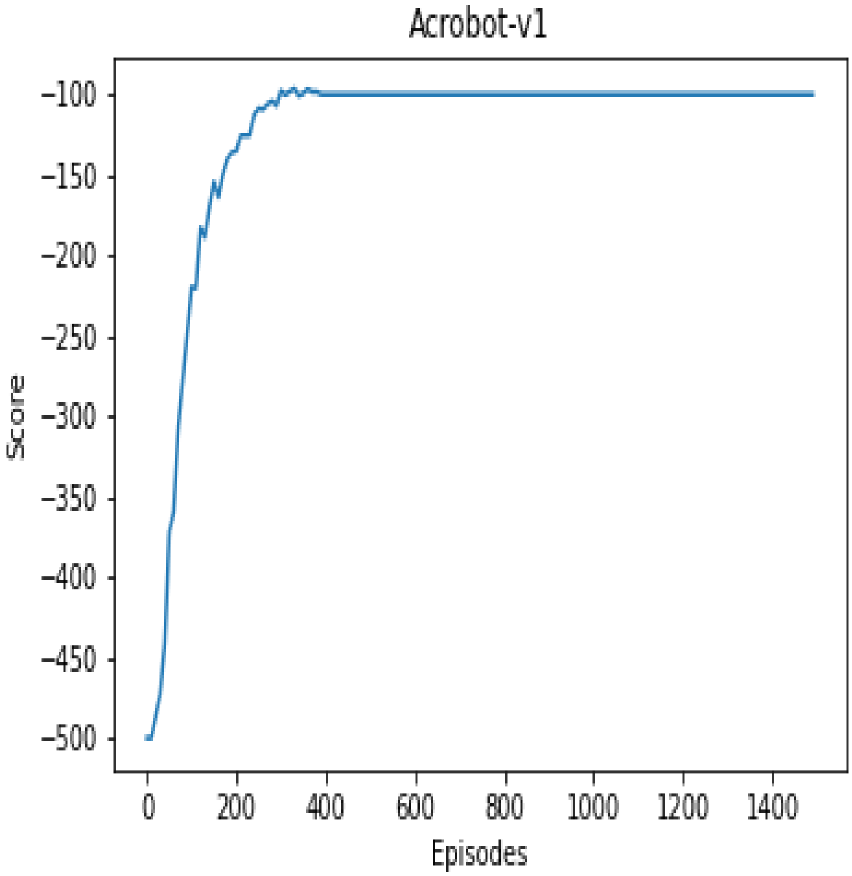
- **Hyper-parameters**
  - Learning Rate: 1e-4
  - Buffer Size: 1e5
  - Update Frequency: 5
  - Batch Size: 64
  - Architecture: 128 - 64

Results:
Average Episodes to converge = 333.8, Convergence Rate = 100%



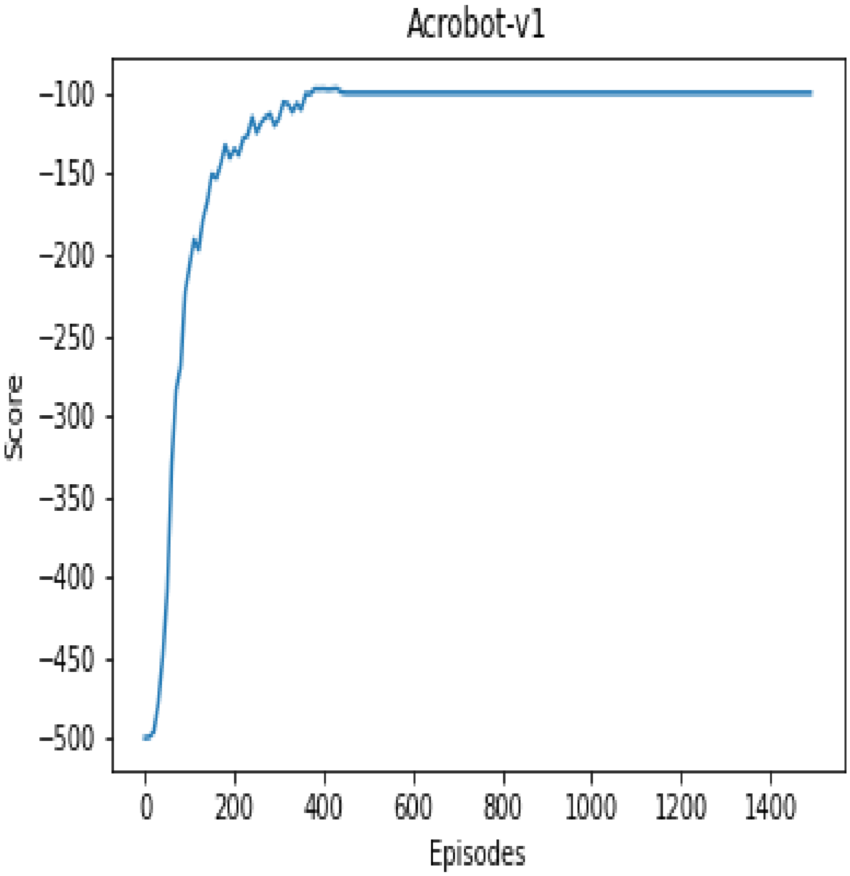
- **Hyper-parameters**
  - Learning Rate: 5e-4
  - Update Frequency: 25
  - Batch Size: 128
  - Buffer Size: 1e5
  - Architecture: 256 - 128

Results:
Average Episodes to converge = 272.8, Convergence Rate = 100%



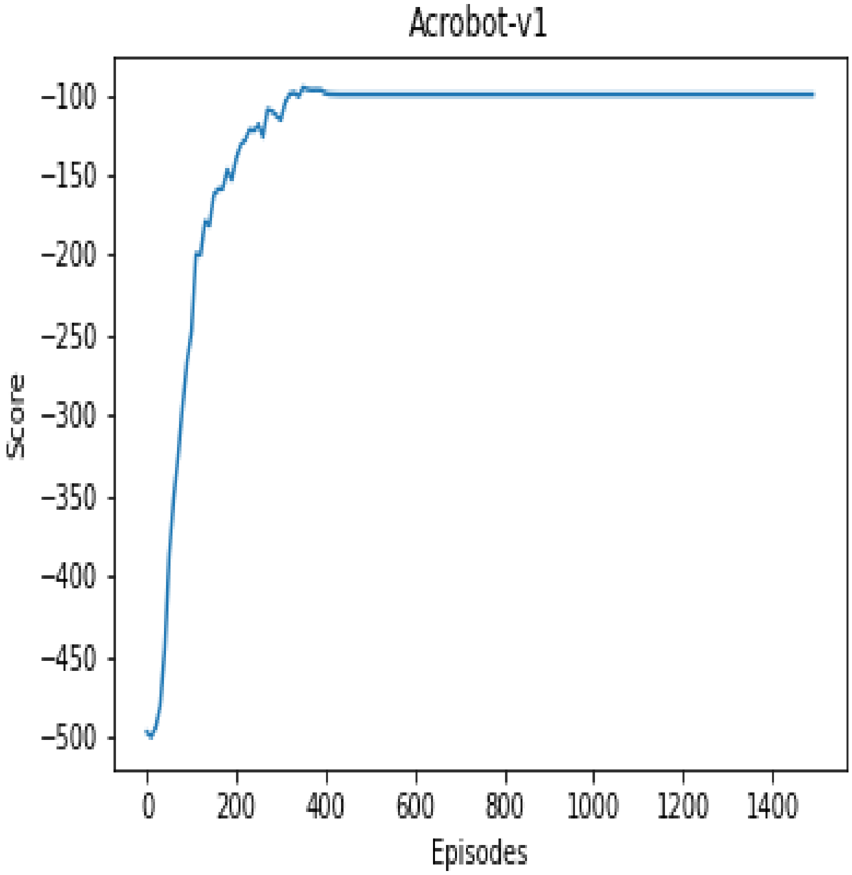
- **Hyper-parameters**
  - Learning Rate: 1e-4
  - Update Frequency: 20
  - Batch Size: 128
  - Buffer Size: 1e5
  - Architecture: 128 - 128

Results:
Average Episodes to converge = 311.2, Convergence Rate = 100%



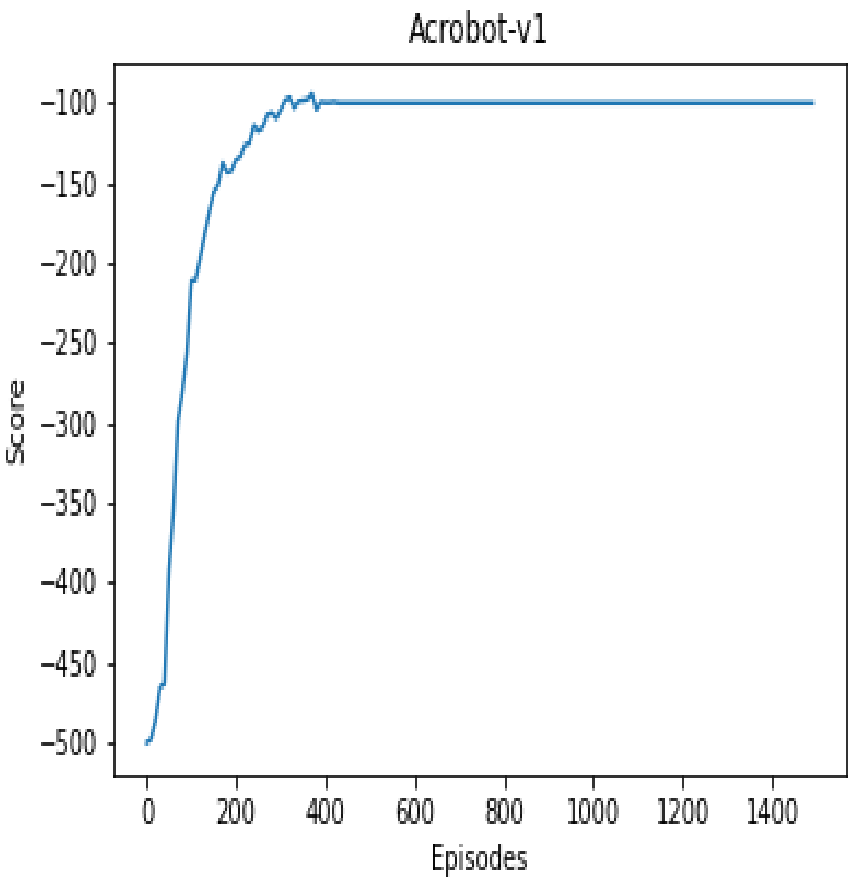
- **Hyper-parameters**
  - Learning Rate: 1e-3
  - Update Frequency: 20
  - Batch Size: 256
  - Buffer Size: 1e5
  - Architecture: 256 - 128

Results:
Average Episodes to converge = 287, Convergence Rate = 100%



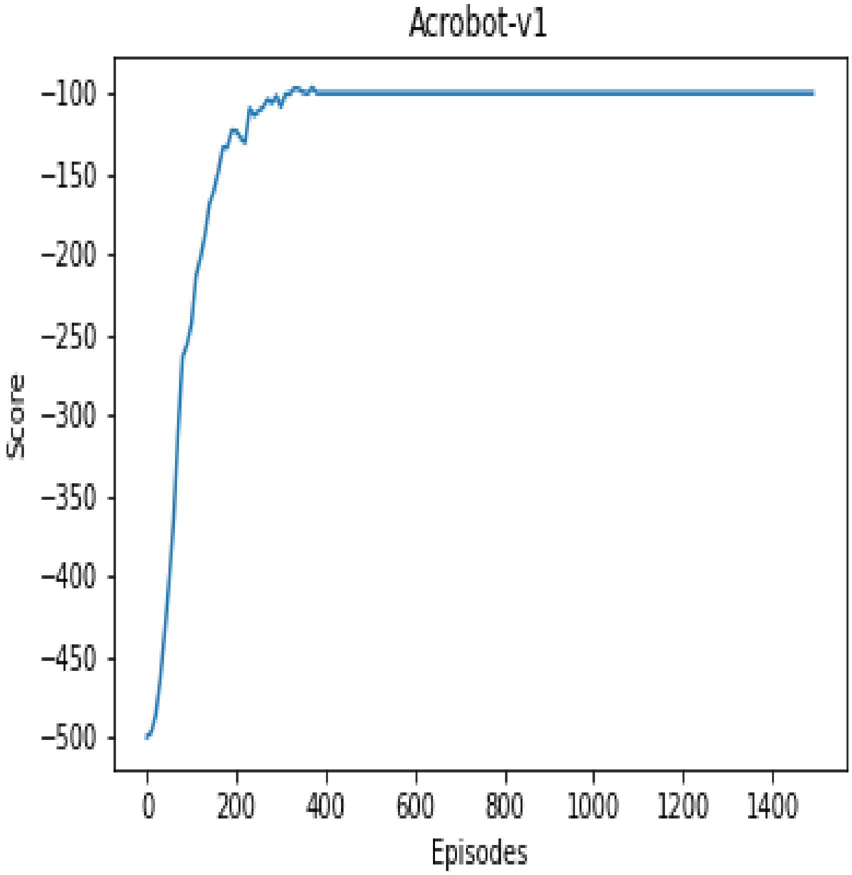
- **Hyper-parameters**
  - Learning Rate: 5e-4
  - Update Frequency: 10
  - Batch Size: 256
  - Buffer Size: 1e3
  - Architecture: 256 - 128

Results:
Average Episodes to converge = 286, Convergence Rate = 100%



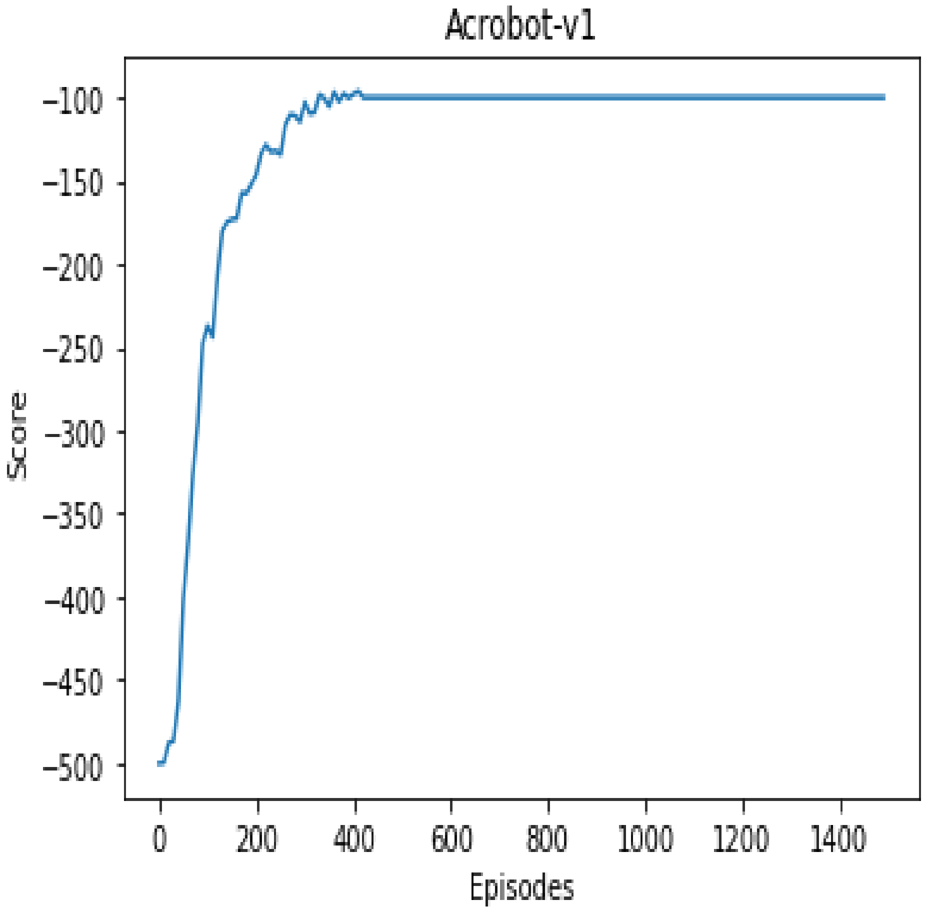
- **Hyper-parameters**
  - Learning Rate: 1e-4
  - Update Frequency: 50
  - Batch Size: 256
  - Buffer Size: 1e5
  - Architecture: 256 - 128

Results:
Average Episodes to converge = 276.8, Convergence Rate = 100%



- **Hyper-parameters**
  - Learning Rate: 5e-4
  - Update Frequency: 20
  - Batch Size: 256
  - Buffer Size: 1e5
  - Architecture: 256 - 256

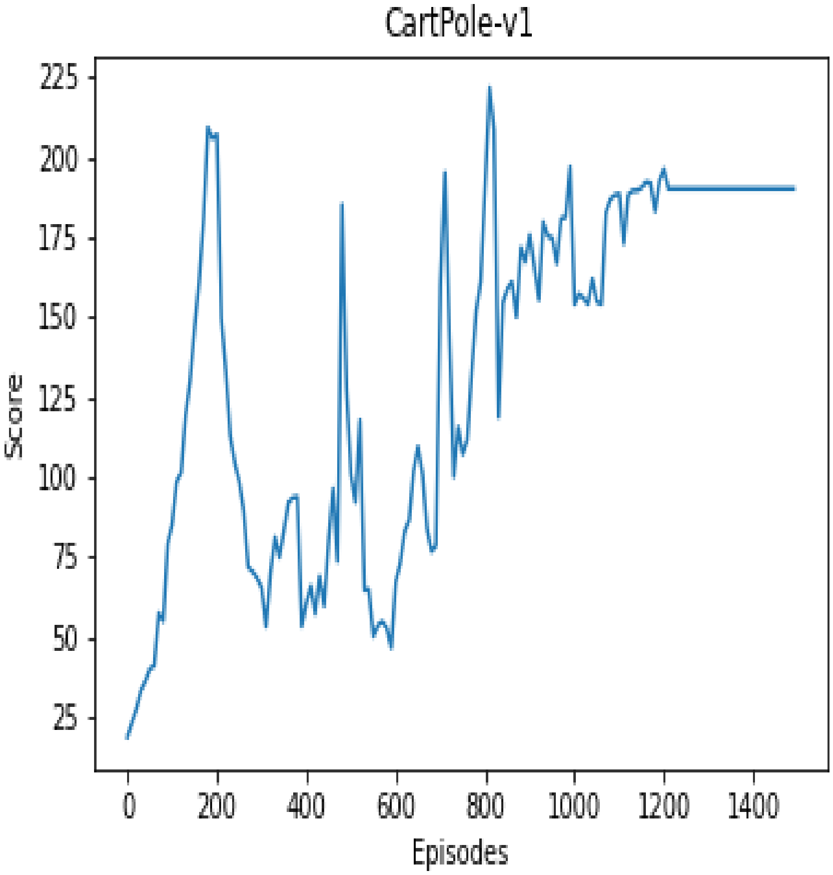
Results:
Average Episodes to converge = 294.6, Convergence Rate = 100%



CartPole

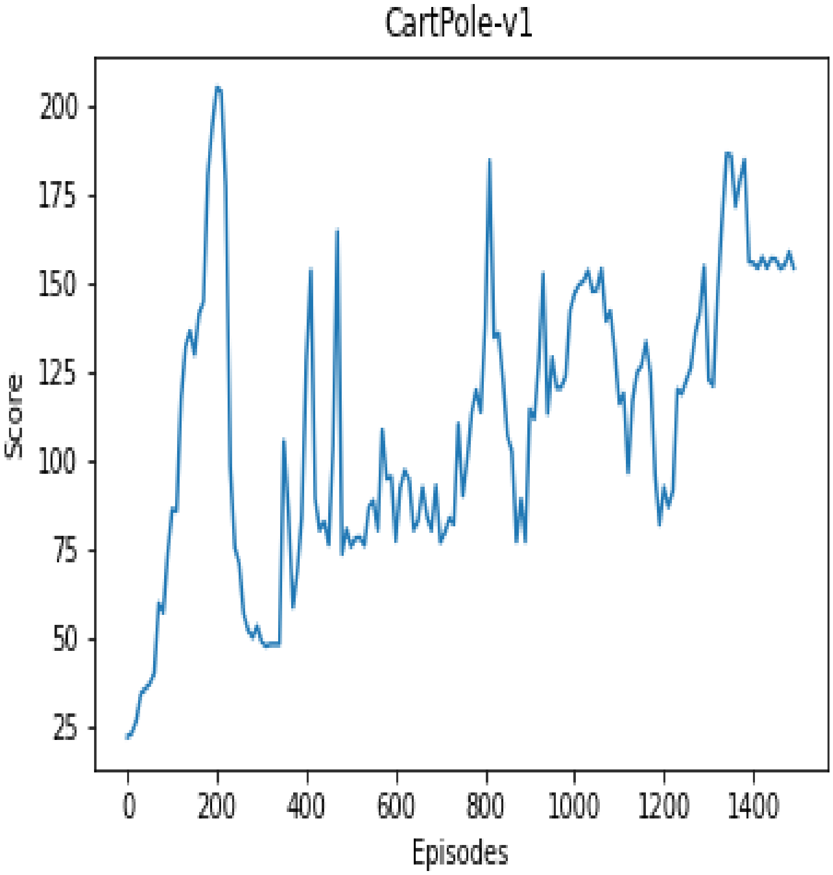
- **Hyper-parameters**
  - Learning Rate: 5e-4
  - Update Frequency: 20
  - Batch Size: 64
  - Buffer Size: 1e5
  - Architecture: 128 - 64

Results:
Average Episodes to converge = 712, Convergence Rate = 100%



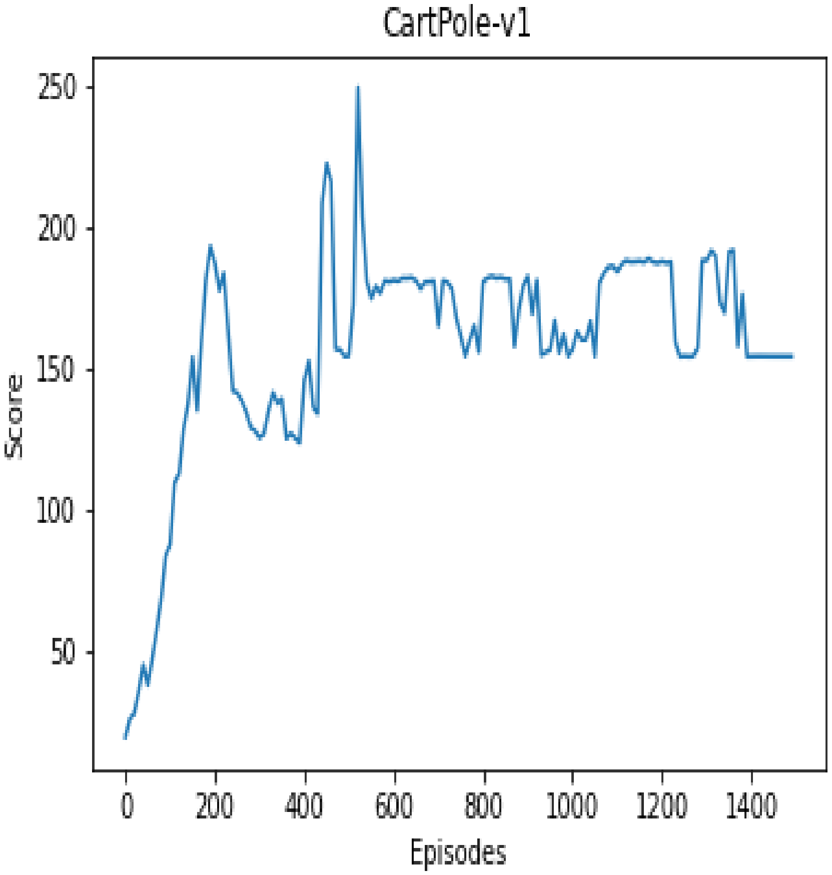
- **Hyper-parameters**
  - Learning Rate: 1e-3
  - Update Frequency: 20
  - Batch Size: 64
  - Buffer Size: 1e5
  - Architecture: 128 - 64

Results:
Average Episodes to converge = 972, Convergence Rate = 80%



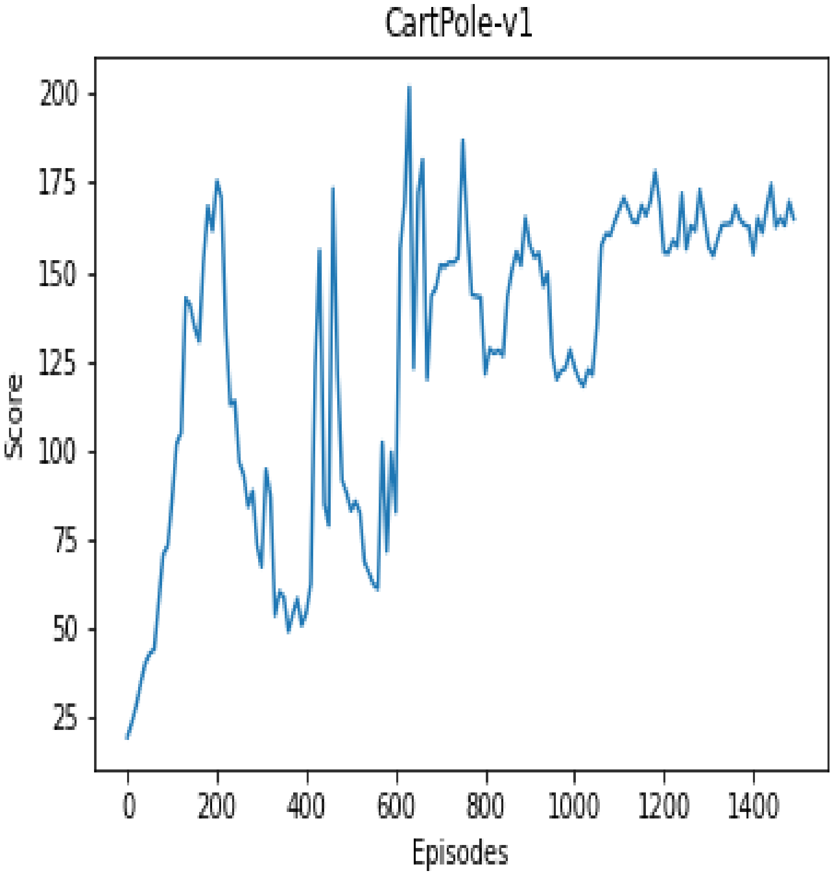
- **Hyper-parameters**
  - Learning Rate: 5e-4
  - Update Frequency: 25
  - Batch Size: 128
  - Buffer Size: 1e5
  - Architecture: 128 - 64

Results:
Average Episodes to converge = 430.8, Convergence Rate = 80%



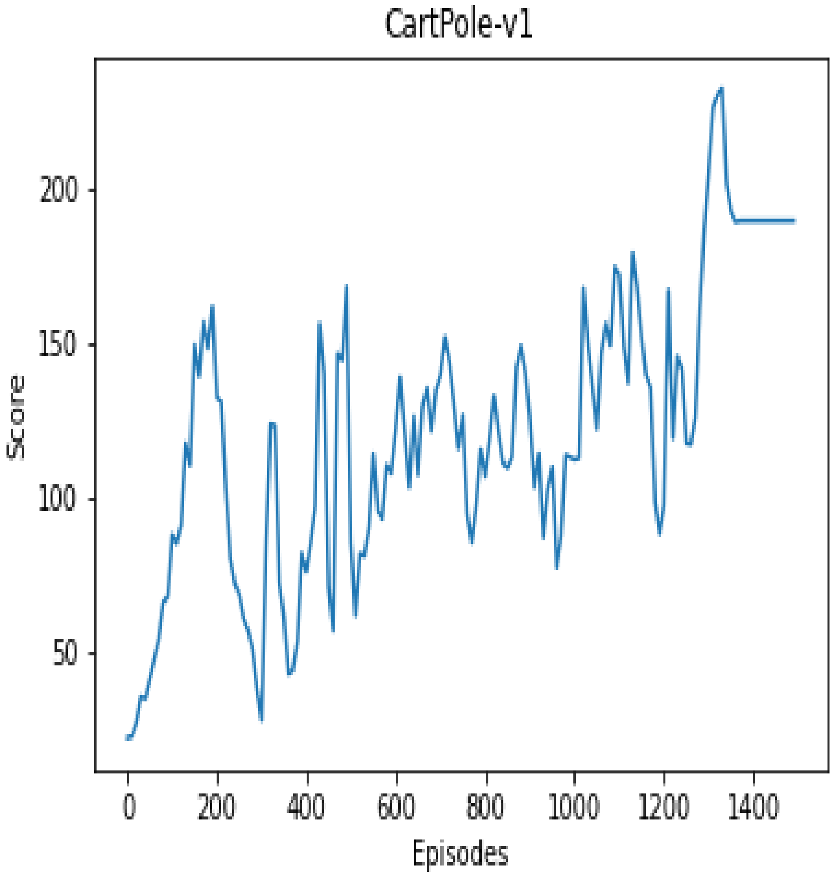
- **Hyper-parameters**
  - Learning Rate: 5e-4
  - Buffer Size: 1e3
  - Update Frequency: 50
  - Batch Size: 64
  - Architecture: 128 - 64

Results:
Average Episodes to converge = 738.2, Convergence Rate = 80%



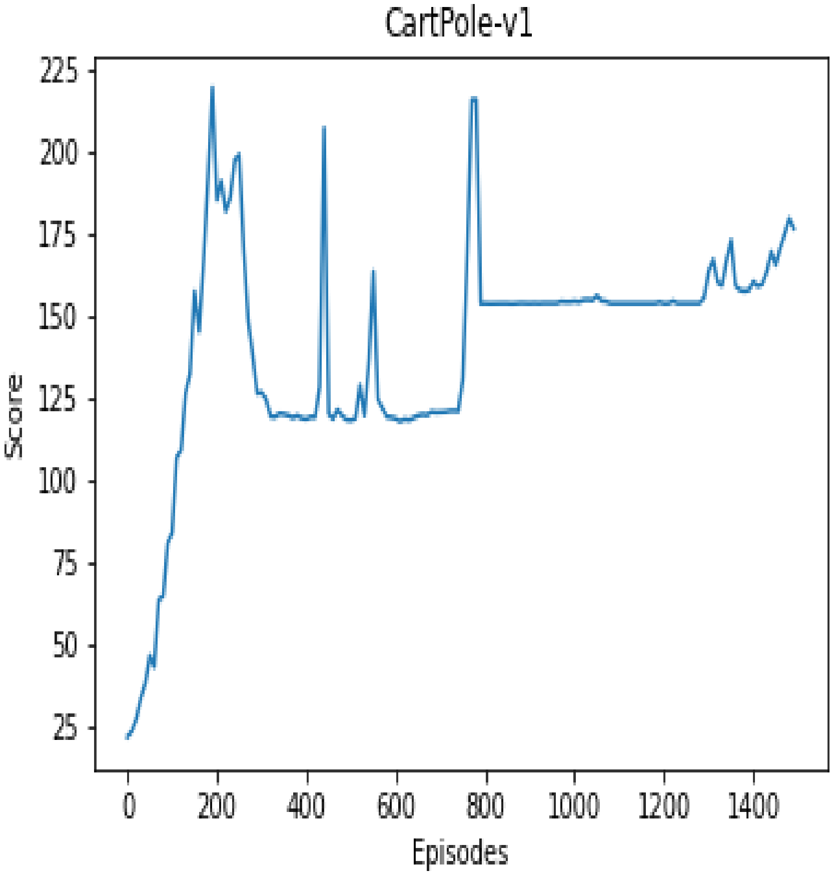
- **Hyper-parameters**
  - Learning Rate: 5e-4
  - Buffer Size: 1e7
  - Update Frequency: 20
  - Batch Size: 32
  - Architecture: 64 - 64

Results:
Average Episodes to converge = 1076.4, Convergence Rate = 80%



- **Hyper-parameters**
  - Learning Rate: 1e-4
  - Update Frequency: 5
  - Batch Size: 64
  - Buffer Size: 1e5
  - Architecture: 128 - 64

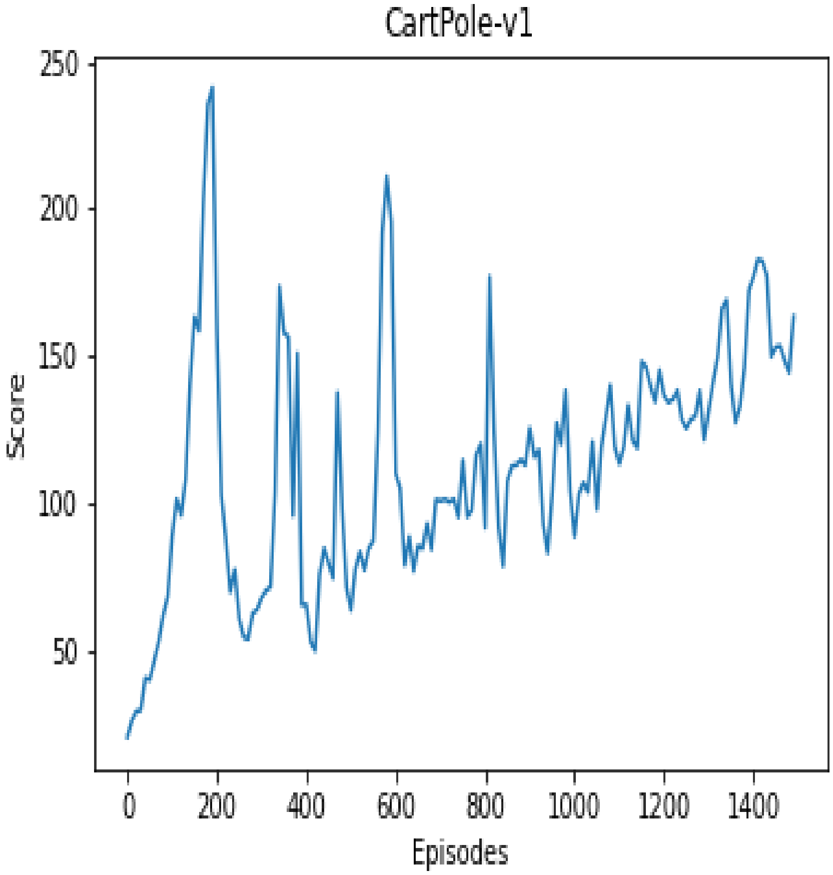
Results:
Average Episodes to converge = 497, Convergence Rate = 80%



- **Hyper-parameters**
  - Learning Rate: 5e-4
  - Update Frequency: 25
  - Batch Size: 128
  - Buffer Size: 1e5
  - Architecture: 256 - 128

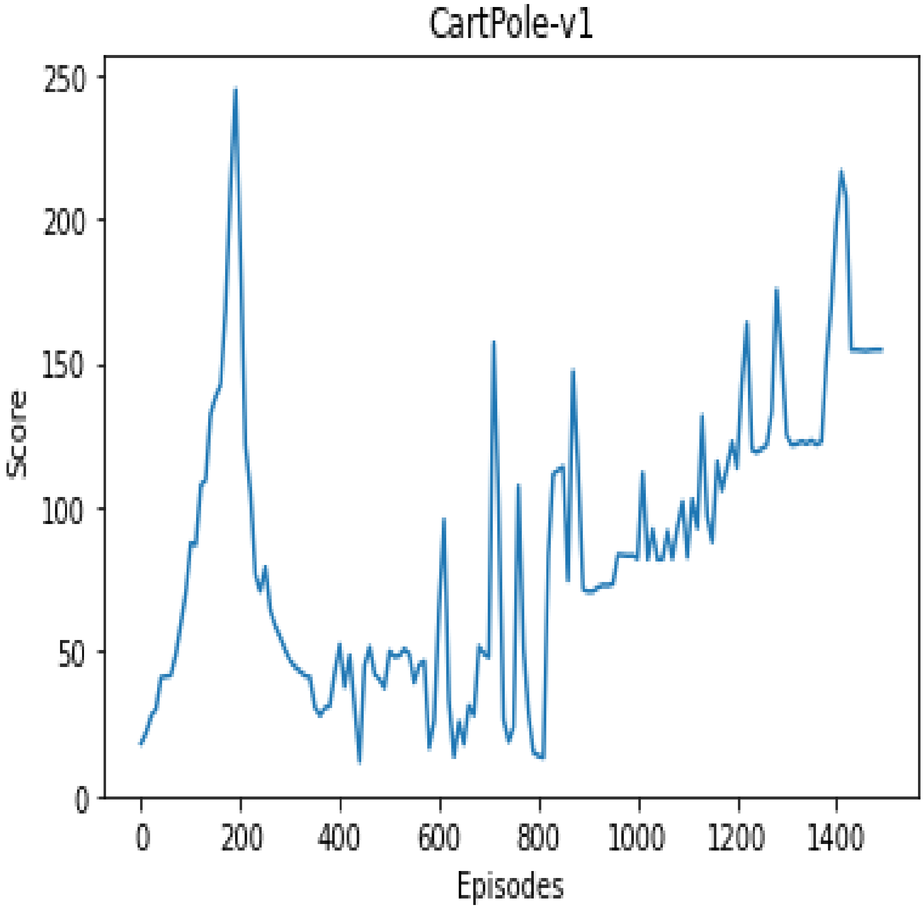
Results:

Average Episodes to converge = 1086.4, Convergence Rate = 60%



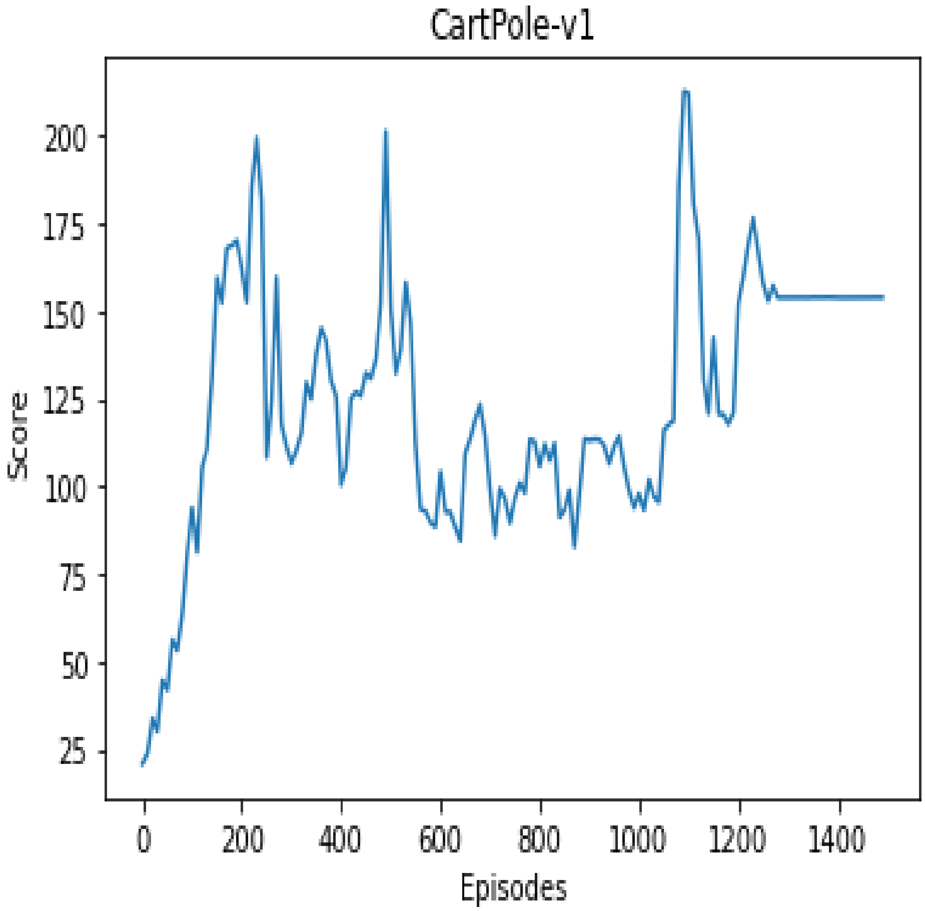
- **Hyper-parameters**
  - Learning Rate: 5e-4
  - Update Frequency: 20
  - Batch Size: 256
  - Buffer Size: 1e5
  - Architecture: 128 - 128

Results:
Average Episodes to converge = 1097.6, Convergence Rate = 80%



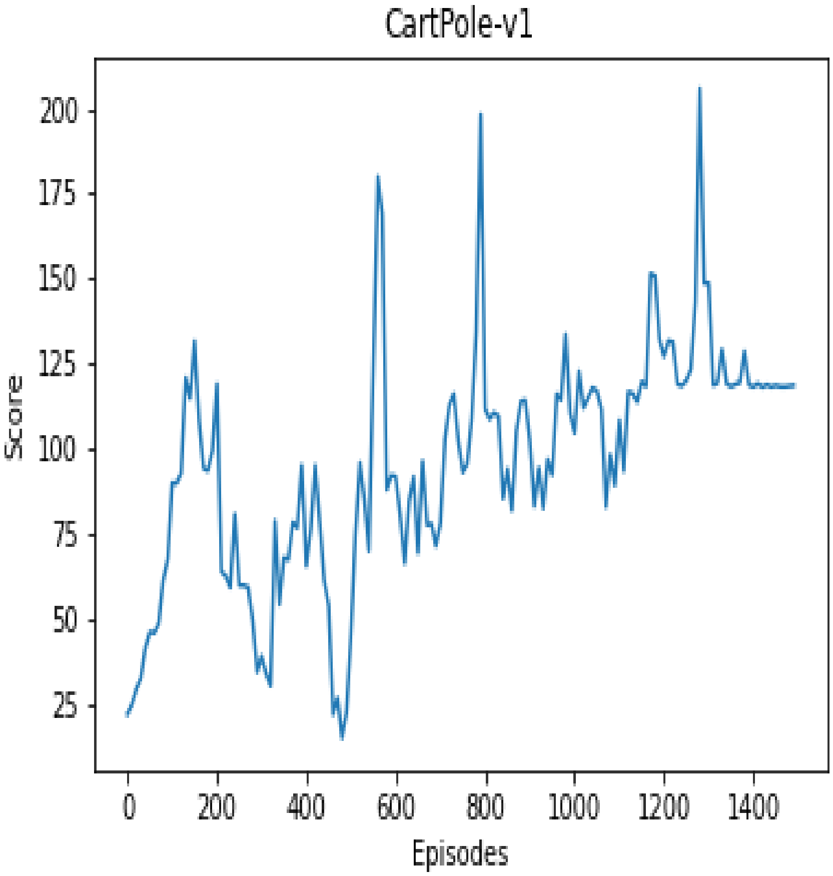
- **Hyper-parameters**
  - Learning Rate: 5e-4
  - Buffer Size: 1e5
  - Update Frequency: 20
  - Batch Size: 256
  - Architecture: 512 - 256

Results:
Average Episodes to converge = 786.4, Convergence Rate = 80%



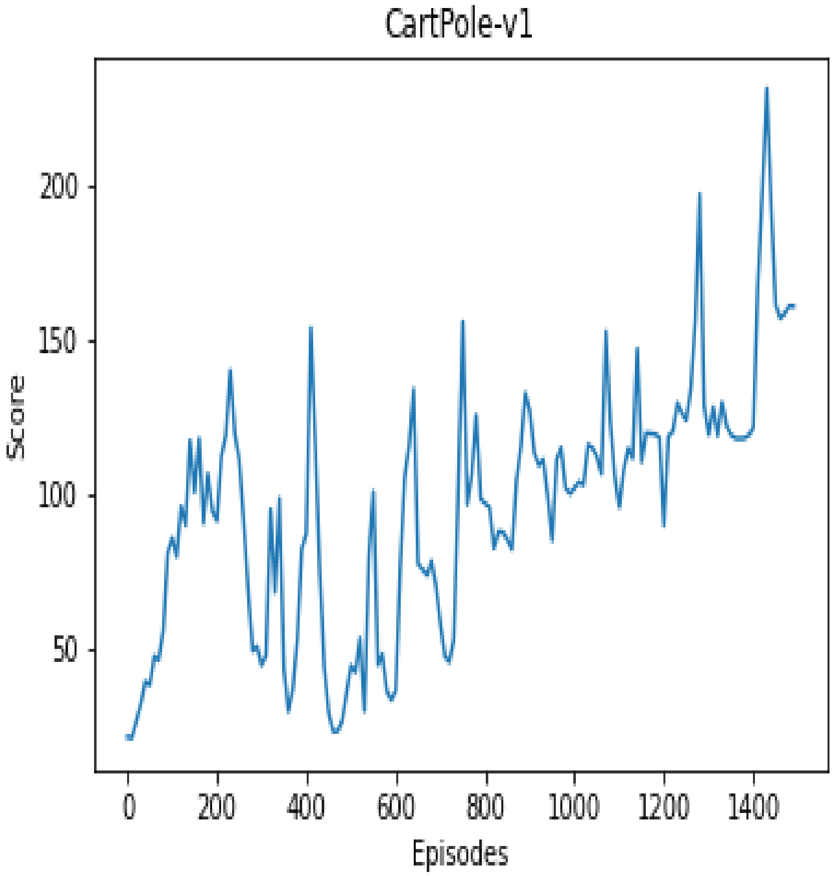
- **Hyper-parameters**
  - Learning Rate: 5e-4
  - Update Frequency: 25
  - Batch Size: 128
  - Buffer Size: 1e5
  - Architecture: 128 - 64 - 32

Results:
Average Episodes to converge = 1020, Convergence Rate = 60%



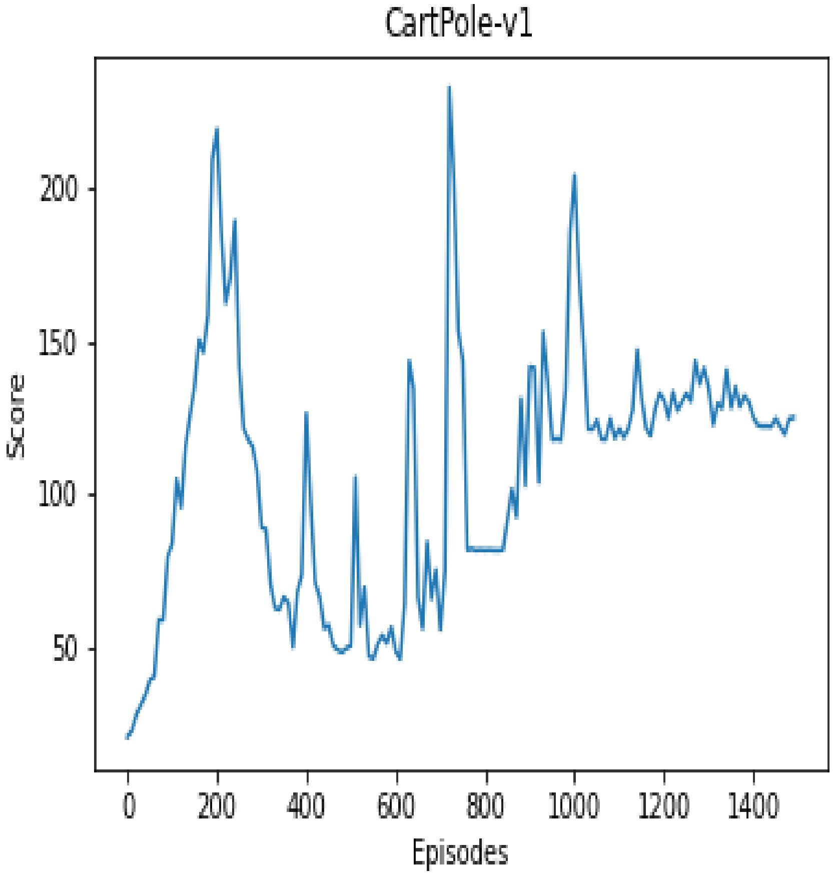
- **Hyper-parameters**
  - Learning Rate: 1e-4
  - Update Frequency: 5
  - Batch Size: 64
  - Buffer Size: 1e5
  - Architecture: 128 - 64 - 32

Results:
Average Episodes to converge = 1044.6, Convergence Rate = 80%



- **Hyper-parameters**
  - Learning Rate: 1e-4
  - Update Frequency: 10
  - Batch Size: 128
  - Buffer Size: 1e3
  - Architecture: 128 - 64

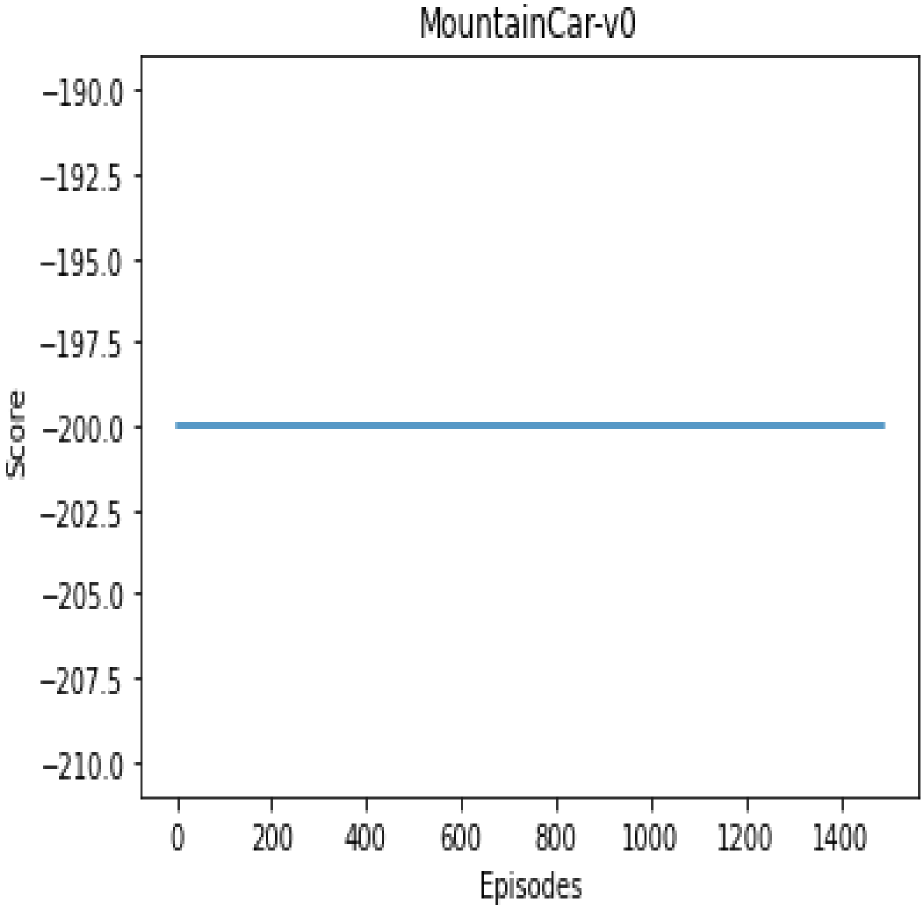
Results:
Average Episodes to converge = 887.4, Convergence Rate = 60%



MountainCar

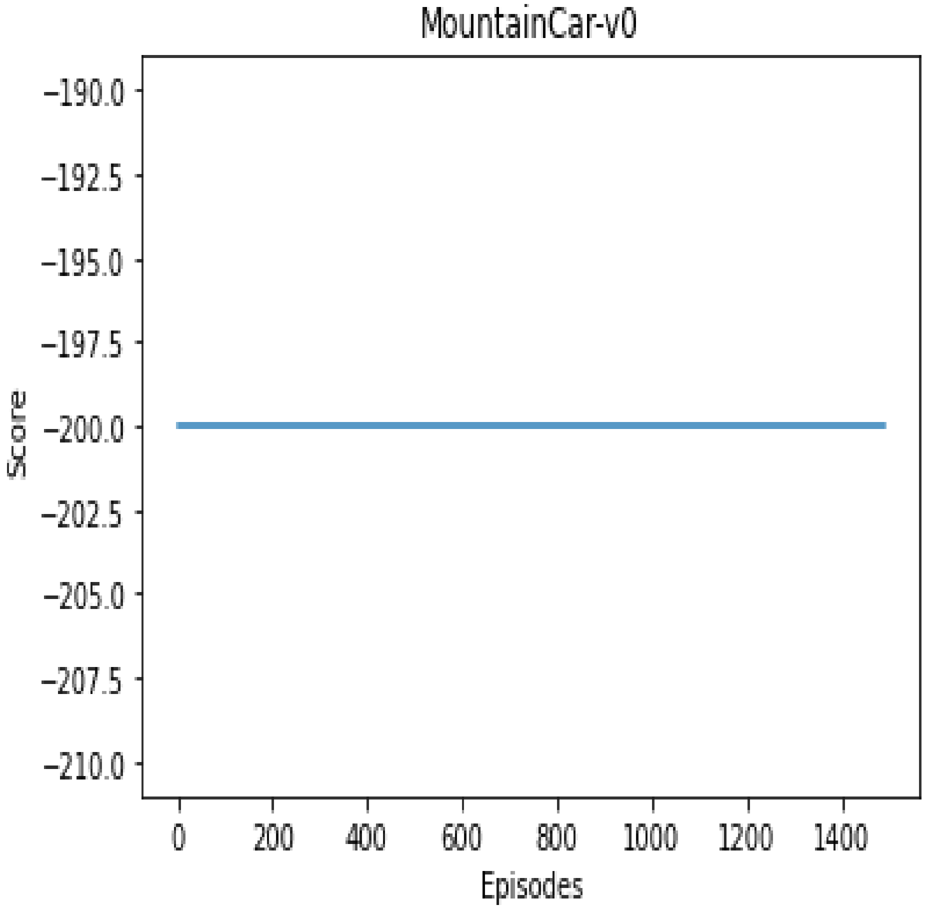
- **Hyper-parameters**
  - Learning Rate: 5e-4
  - Buffer Size: 1e5
  - Update Frequency: 20
  - Batch Size: 64
  - Architecture: 128 - 64

Results:
Average Episodes to converge = 1500, Convergence Rate = 0%



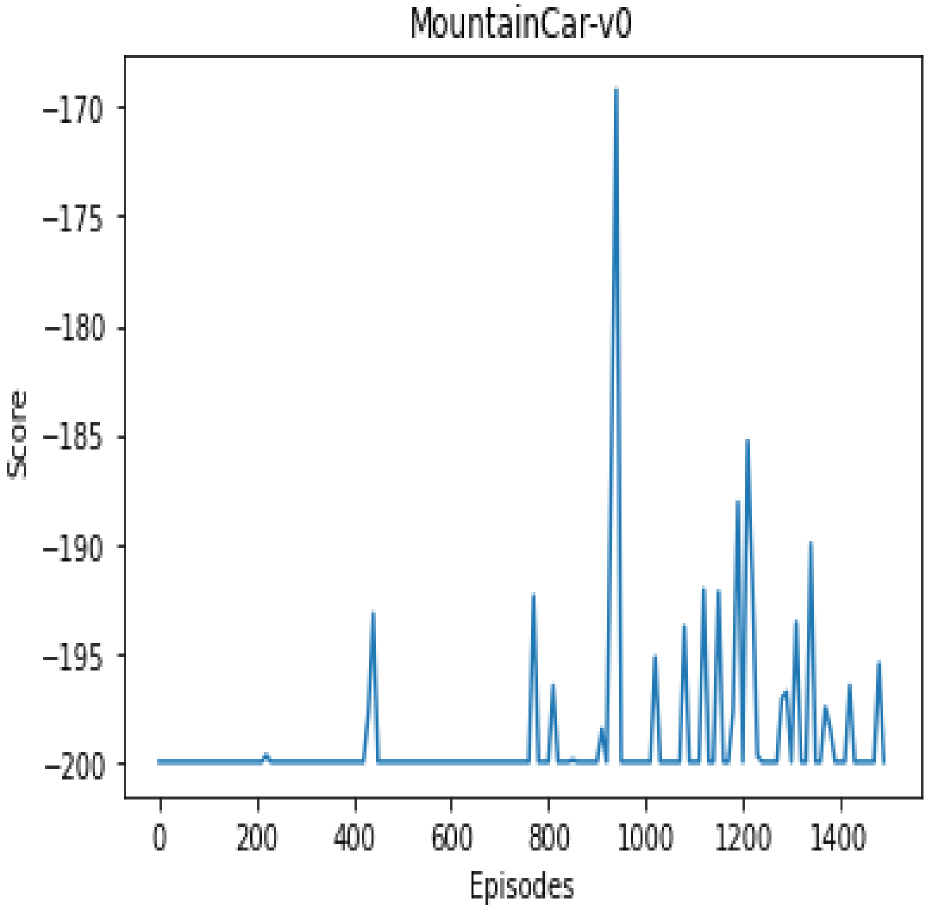
- **Hyper-parameters**
  - Learning Rate: 1e-3
  - Update Frequency: 20
  - Batch Size: 64
  - Buffer Size: 1e5
  - Architecture: 256 - 256

Results:
Average Episodes to converge = 1500, Convergence Rate = 0%



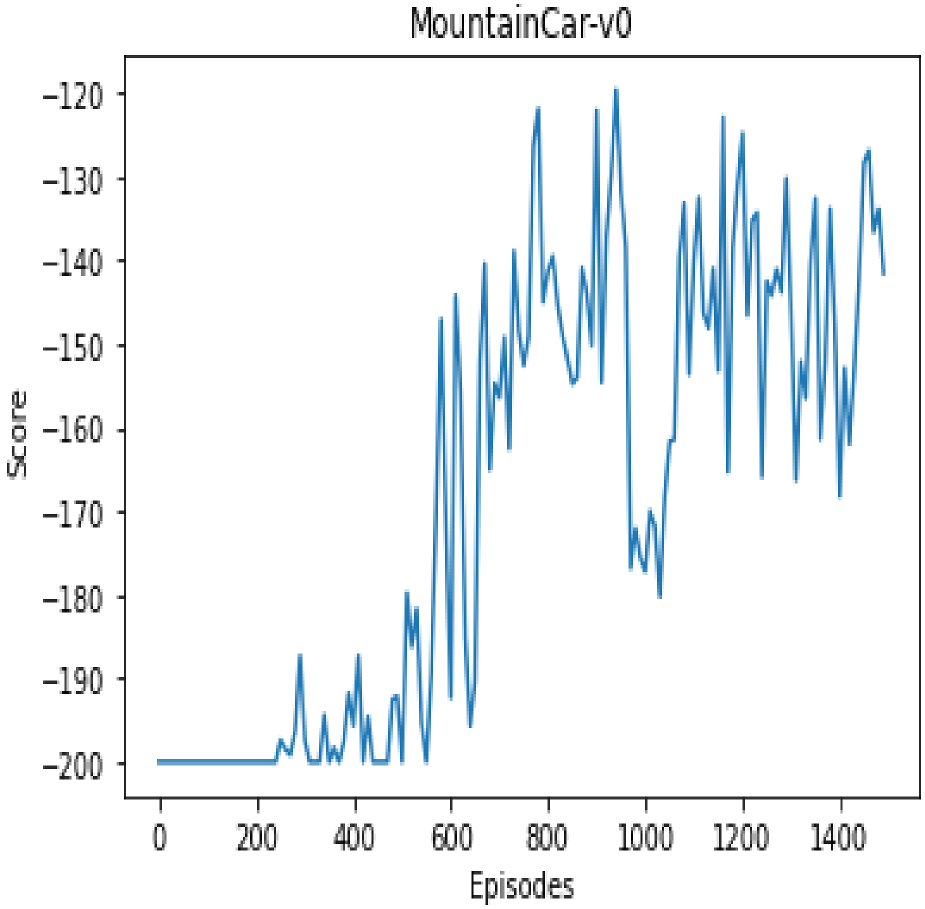
- **Hyper-parameters**
  - Learning Rate: 1e-2
  - Update Frequency: 20
  - Batch Size: 64
  - Buffer Size: 1e5
  - Architecture: 128 - 64

Results:
Average Episodes to converge = 1500, Convergence Rate = 0%



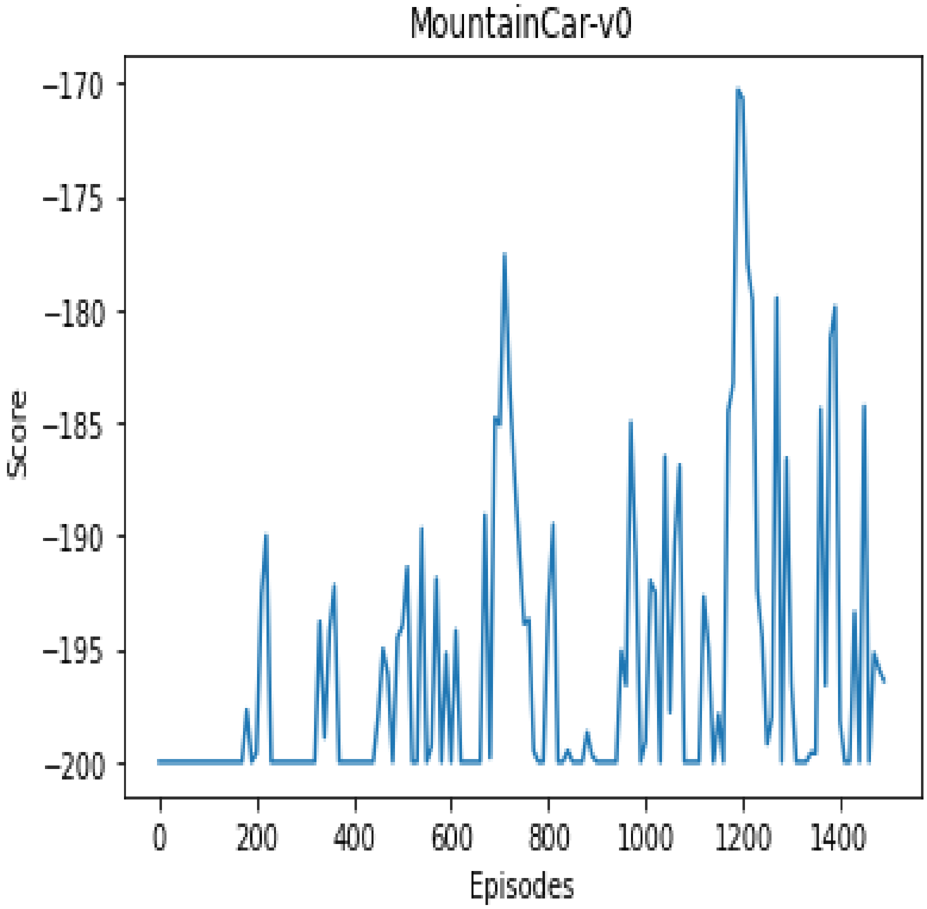
- **Hyper-parameters**
  - Learning Rate: 1e-3
  - Update Frequency: 20
  - Batch Size: 64
  - Buffer Size: 1e5
  - Architecture: 128 - 64 - 64

Results:
Average Episodes to converge = 1423.2, Convergence Rate = 20%



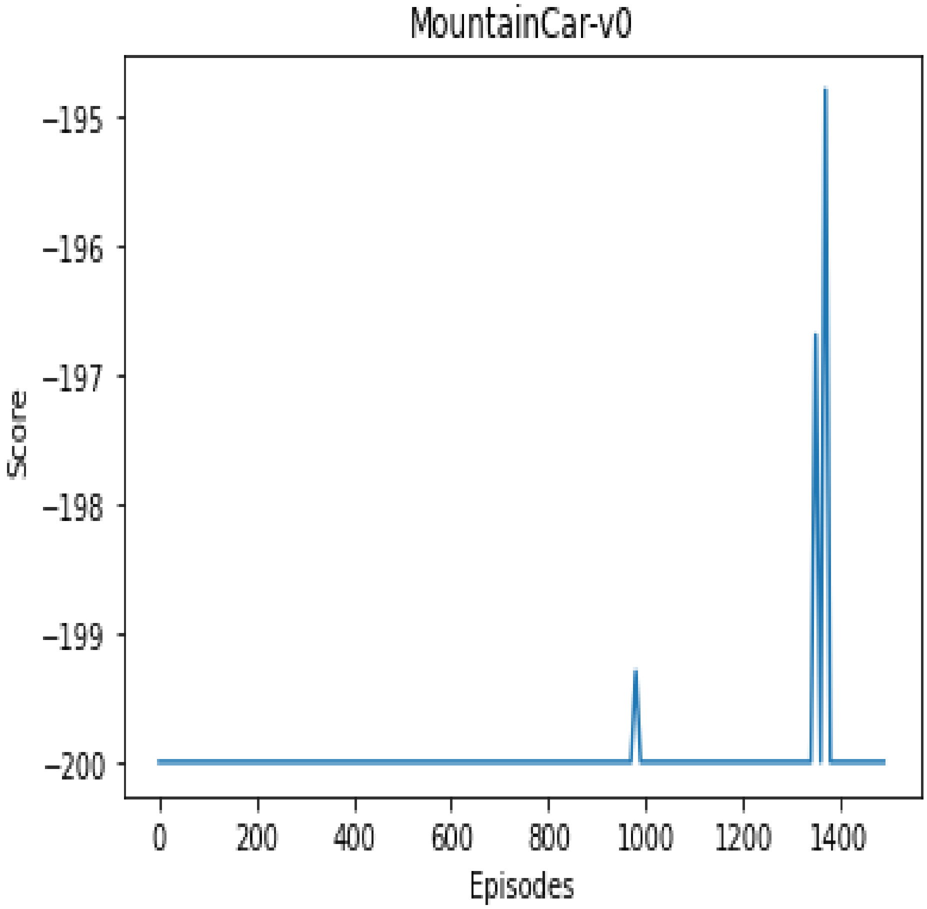
- **Hyper-parameters**
  - Learning Rate: 1e-2
  - Update Frequency: 20
  - Batch Size: 64
  - Buffer Size: 1e5
  - Architecture: 256 - 128 - 64 - 32

Results:
Average Episodes to converge = 1500, Convergence Rate = 0%



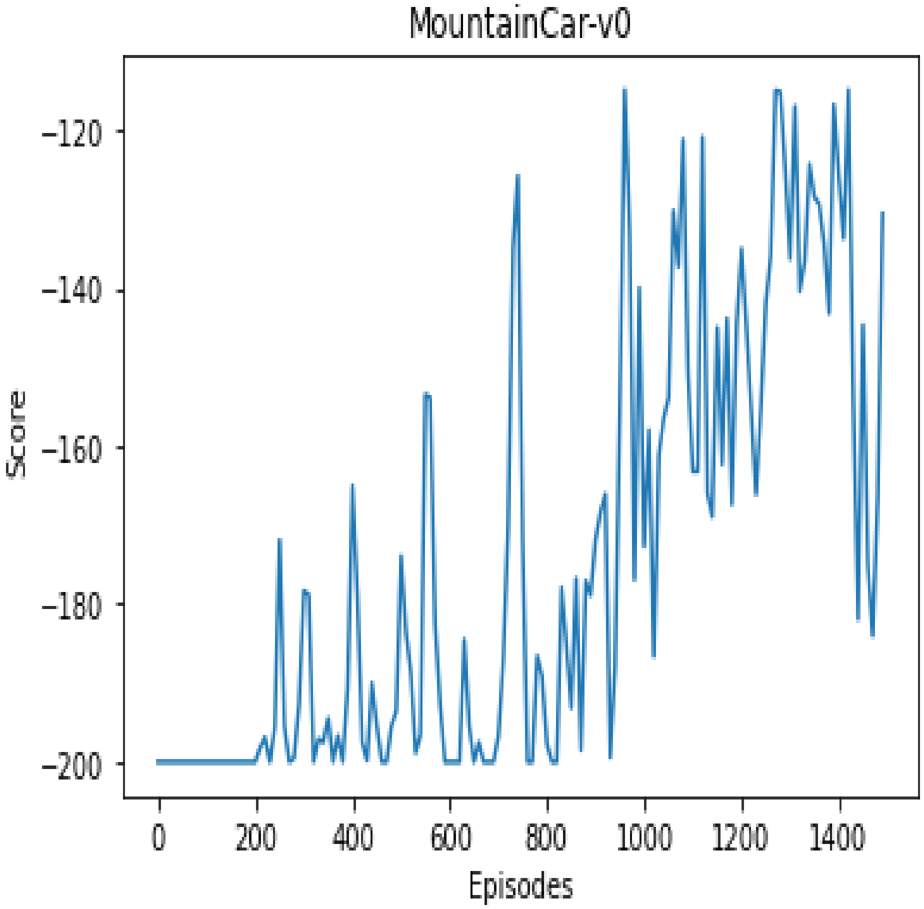
- **Hyper-parameters**
  - Learning Rate: 1e-3
  - Update Frequency: 20
  - Batch Size: 32
  - Buffer Size: 1e5
  - Architecture: 256 - 128 - 64

Results:
Average Episodes to converge = 1500, Convergence Rate = 0%



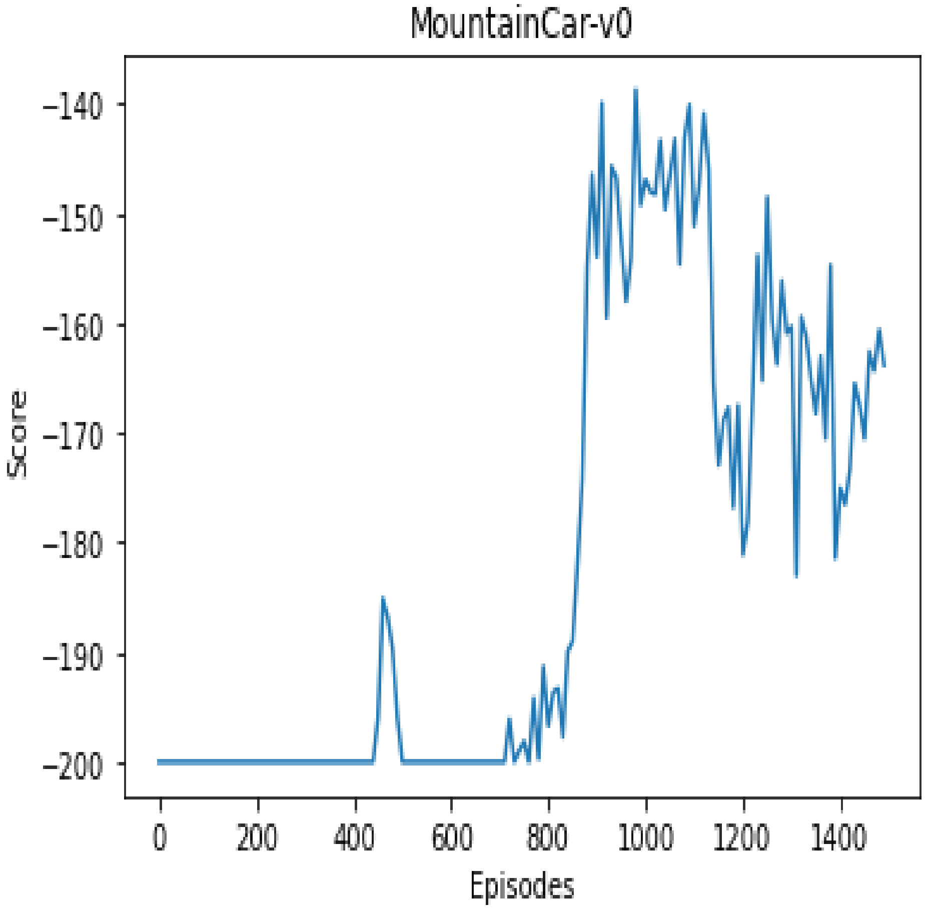
- **Hyper-parameters**
  - Learning Rate: 1e-3
  - Buffer Size: 1e5
  - Update Frequency: 50
  - Batch Size: 64
  - Architecture: 128 - 64 - 64

Results:
Average Episodes to converge = 1452.6, Convergence Rate = 20%



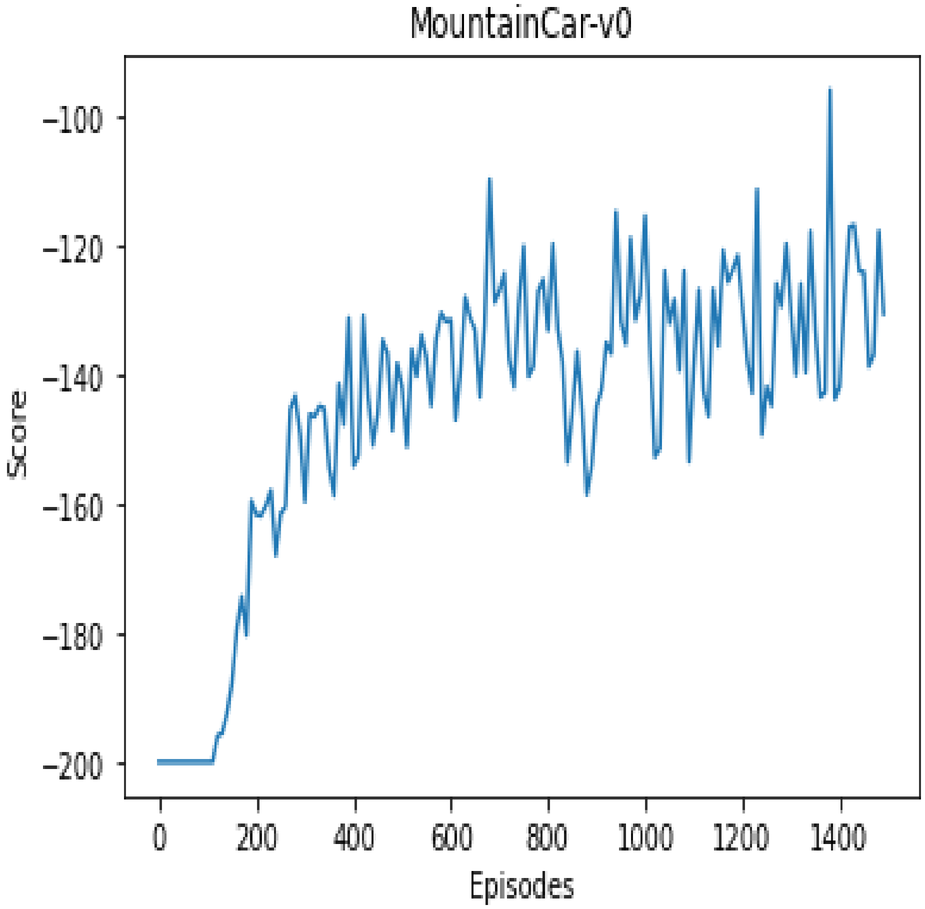
- **Hyper-parameters**
  - Learning Rate: 5e-4
  - Update Frequency: 20
  - Batch Size: 64
  - Buffer Size: 1e5
  - Architecture: 1024

Results:
Average Episodes to converge = 1432.8, Convergence Rate = 20%



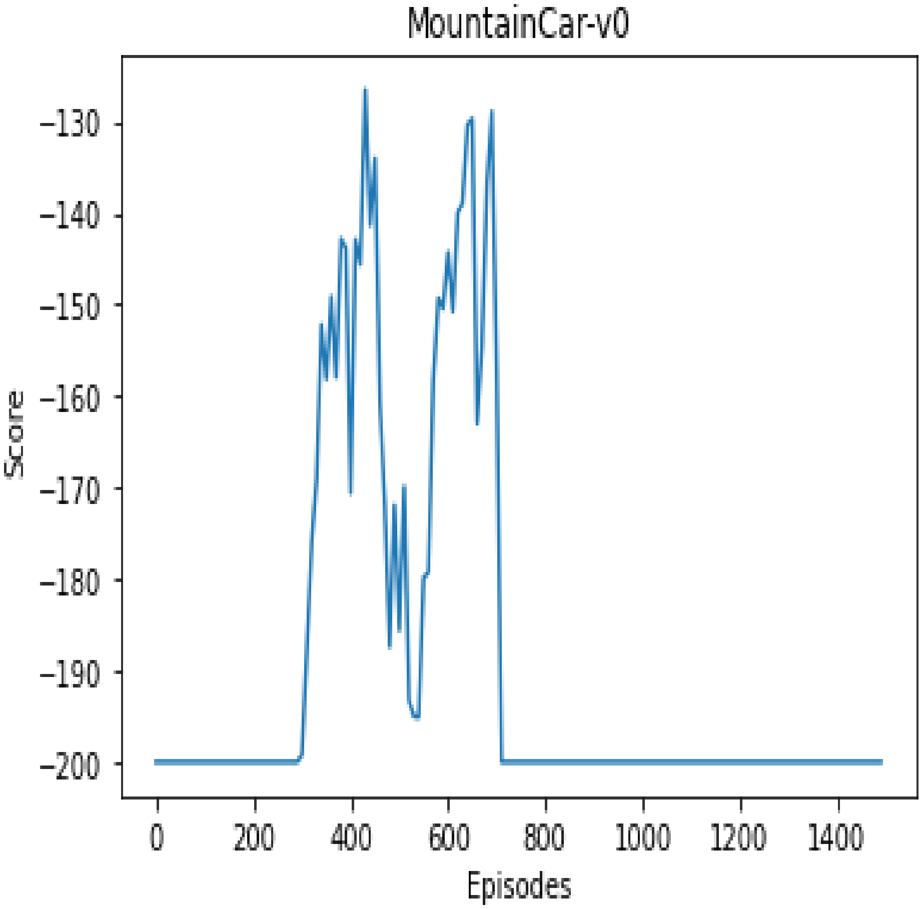
- **Hyper-parameters**
  - Learning Rate: 1e-2
  - Update Frequency: 20
  - Batch Size: 64
  - Buffer Size: 1e5
  - Architecture: 2048

Results:
Average Episodes to converge = 1394.2, Convergence Rate = 40%



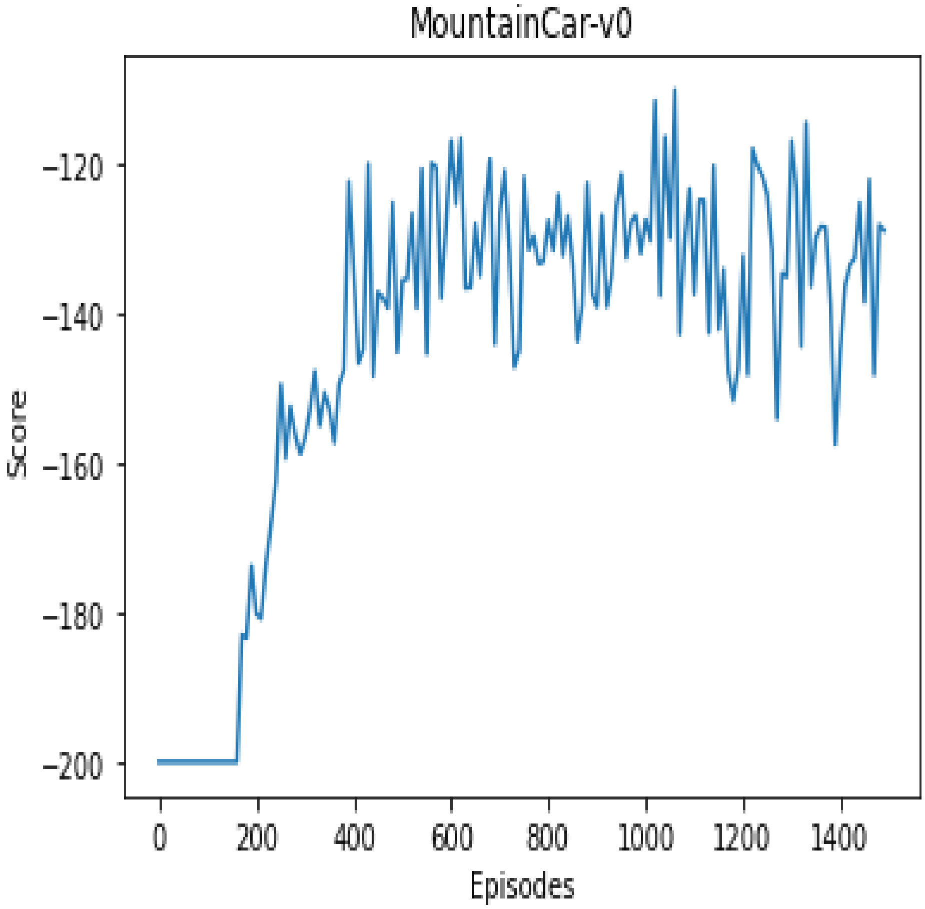
- **Hyper-parameters**
  - Learning Rate: 1e-1
  - Buffer Size: 1e7
  - Update Frequency: 20
  - Batch Size: 64
  - Architecture: 2048

Results:
Average Episodes to converge = 1500, Convergence Rate = 0%



- **Hyper-parameters**
  - Learning Rate: 1e-3
  - Update Frequency: 20
  - Batch Size: 64
  - Buffer Size: 1e3
  - Architecture: 2048

Results:
Average Episodes to converge = 1376.4, Convergence Rate = 0%



3 DQN Inference

Acrobot:

Learning Rate	Update Frequency	Batch Size	Buffer Size	FC1	FC2	Average Episodes
5.00E-04	20	64	1.00E+05	128	64	300.6
1.00E-03	20	64	1.00E+05	128	64	350.2
5.00E-04	25	128	1.00E+05	128	64	300.6
5.00E-04	50	64	1.00E+03	128	64	343.8
5.00E-04	20	32	1.00E+07	64	64	350.6
1.00E-04	5	64	1.00E+05	128	64	333.8
5.00E-04	25	128	1.00E+05	256	128	272.8
1.00E-04	20	128	1.00E+05	128	128	311.2
1.00E-03	20	256	1.00E+05	256	128	287
5.00E-04	10	256	1.00E+03	256	128	286
1.00E-04	50	256	1.00E+05	256	128	276.8
5.00E-04	20	256	1.00E+05	256	256	294.6

- Acrobot was relatively the easiest environment to solve among the 3 environments.
- Batch size had a great impact in the episodes taken to converge. As increasing the batch size decreases stochasticity, which in-turn helps the network to reach the minima faster.
- Increasing the network size had an impact, however after a certain point the improvement started diminishing. Therefore an architecture with 256 - 128 nodes is the sweet spot.

CartPole:

LR	Update Freq	Batch Size	Buffer Size	FC1	FC2	Average Episodes	Coverged
5.00E-04	20	64	1.00E+05	128	64	712	100
1.00E-03	20	64	1.00E+05	128	64	972	80
5.00E-04	25	128	1.00E+05	128	64	430.8	80
5.00E-04	50	64	1.00E+03	128	64	738.2	80
5.00E-04	20	32	1.00E+07	64	64	1076.4	80
1.00E-04	5	64	1.00E+05	128	64	497	80
5.00E-04	25	128	1.00E+05	256	128	1086.4	60
5.00E-04	20	256	1.00E+05	128	128	1097.6	80
1.00E-04	10	128	1.00E+03	128	64	887.4	60
5.00E-04	20	256	1.00E+05	512	256	786.4	80
5.00E-04	25	128	1.00E+05	128	64+32	1020	60
1.00E-04	5	64	1.00E+05	128	64+32	1044.6	80

- For Cartpole, given hyperparameters gave a good result, therefore it was difficult to improve on it further.

- Various architectures were tested out, including a couple of 3 layers based architectures, however bigger architecture tend to increase the regret and also takes a lot more time.

- Increasing the update frequency gave a better result, but an extensive experimentation might be needed to prove this hypothesis.

MountainCar:

LR	Update Freq	Batch Size	Buffer Size	FC1	FC2	Average Episodes	Coverged
5.00E-04	20	64	1.00E+05	128	64	1500	0
5.00E-04	20	64	1.00E+05	256	256	1500	0
1.00E-02	20	64	1.00E+05	128	64	1500	0
1.00E-03	20	64	1.00E+05	128	64+64	1423.2	20
1.00E-02	20	64	1.00E+05	256	128+64+32	1500	0
1.00E-03	20	32	1.00E+05	256	128+64	1500	0
1.00E-03	50	64	1.00E+05	128	64+64	1452.6	20
5.00E-04	20	64	1.00E+05	1024	0	1432.8	20
1.00E-02	20	64	1.00E+05	2048	0	1394.2	40
1.00E-01	20	64	1.00E+07	2048	0	1500	0
1.00E-03	20	64	1.00E+03	2048	0	1376.4	40

- MountainCar was the hardest environment to solve among the 3 environments.

- Deeper architectures was tested out, initially there was an improvement, however much deeper architectures were unable to learn the trick to climb the mountain.

- Rendering the environment made us realise the agent was trying to go faster forward and was not making use of the momentum by going backward.

- We tried shallow network with lot more units, which yield much better results.

- Decreasing buffer size also helped the agent to learn, as there was more randomness and the agent was exploring more.

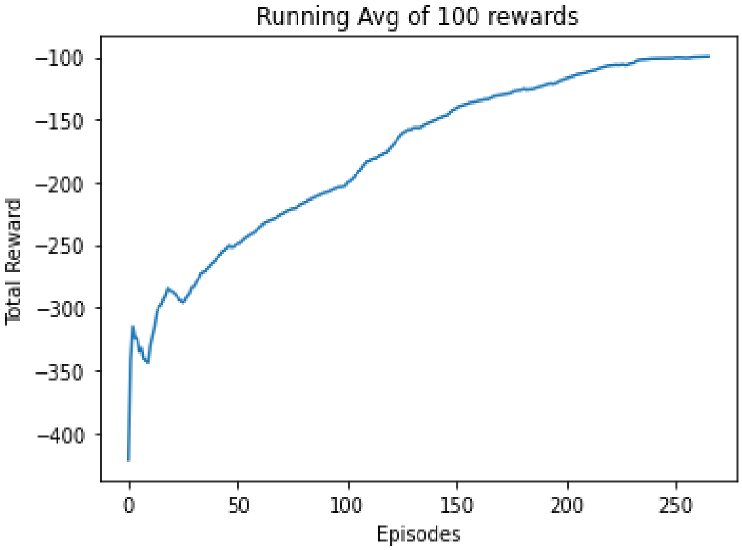
4 Actor-Critic

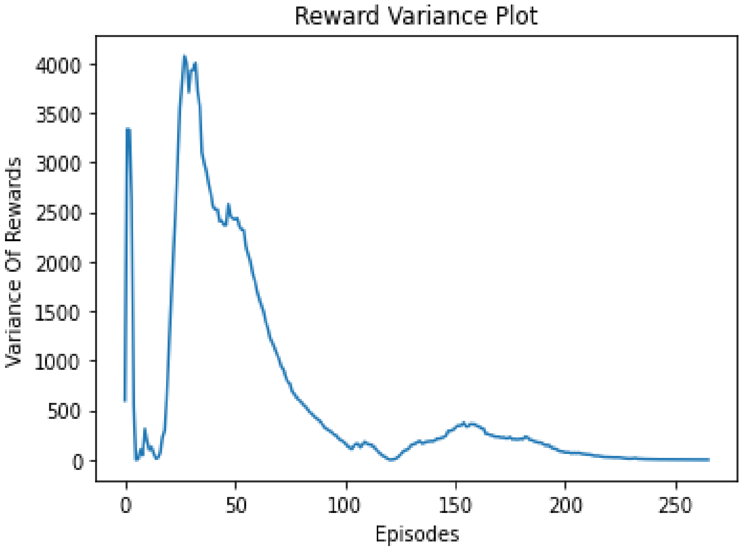
Acrobot

- **Hyper-parameters**
  - Methodology: One Step
  - Learning Rate: 1e-3
  - Architecture: 128 - 64
  - Episodes: 1000

Results:

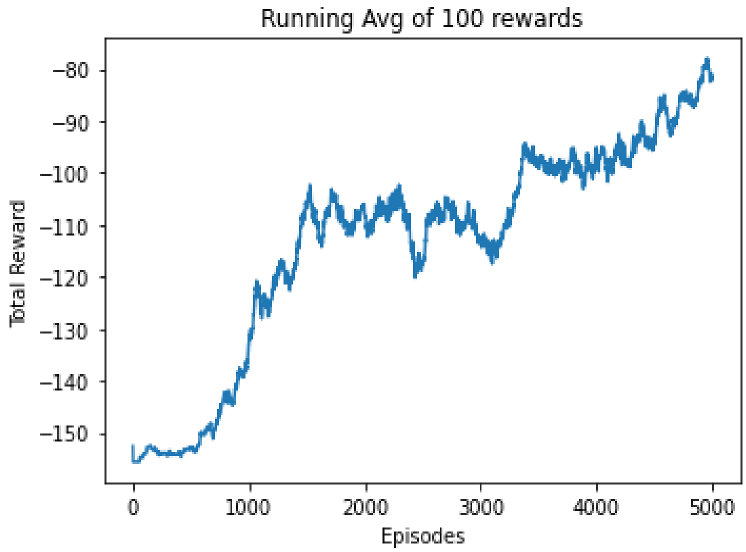
Average Episodes to converge = 166, Convergence Rate = 40%

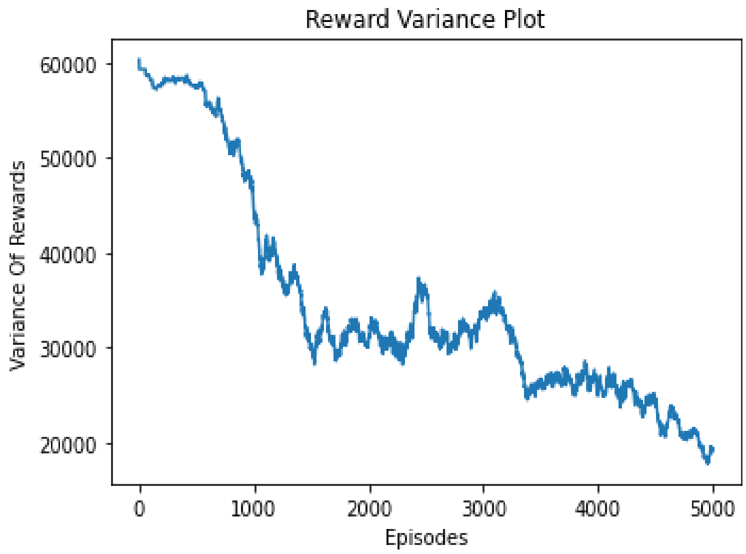




- **Hyper-parameters**
  - Methodology: Full Return
  - Learning Rate: 1e-3
  - Architecture: 128 - 64
  - Episodes: 1000

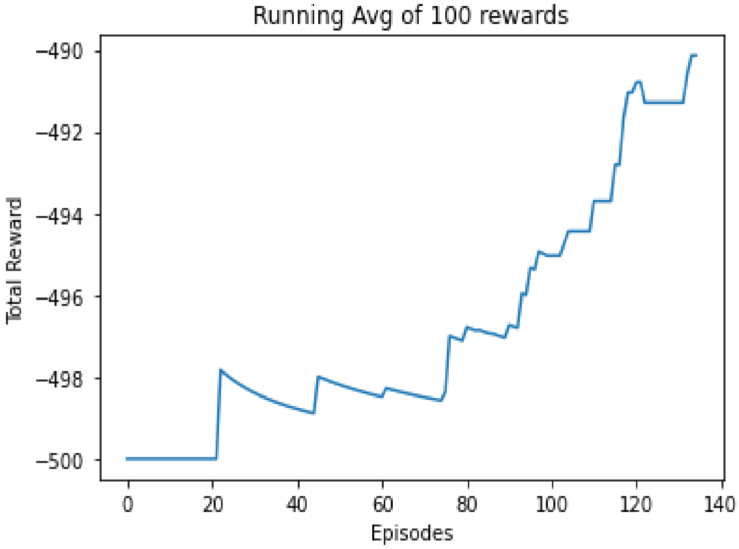
Results:
Average Episodes to converge = 140, Convergence Rate = 35%

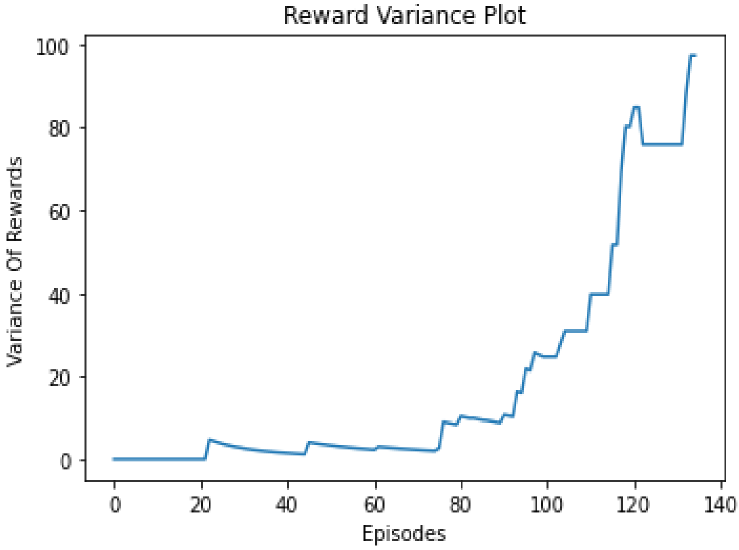




- **Hyper-parameters**
  - Methodology: N-step Return with N = 31
  - Learning Rate: 1e-4
  - Architecture: 1024 - 512
  - Episodes: 1000

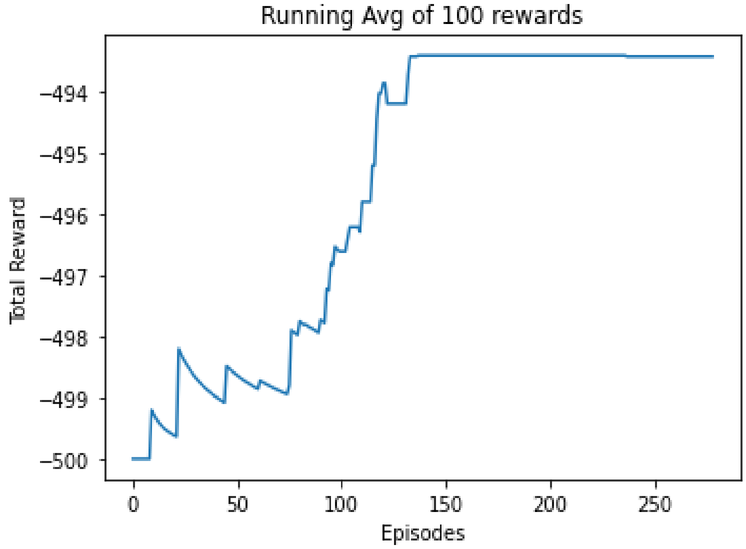
Results:
Average Episodes to converge = 140, Convergence Rate = 35%

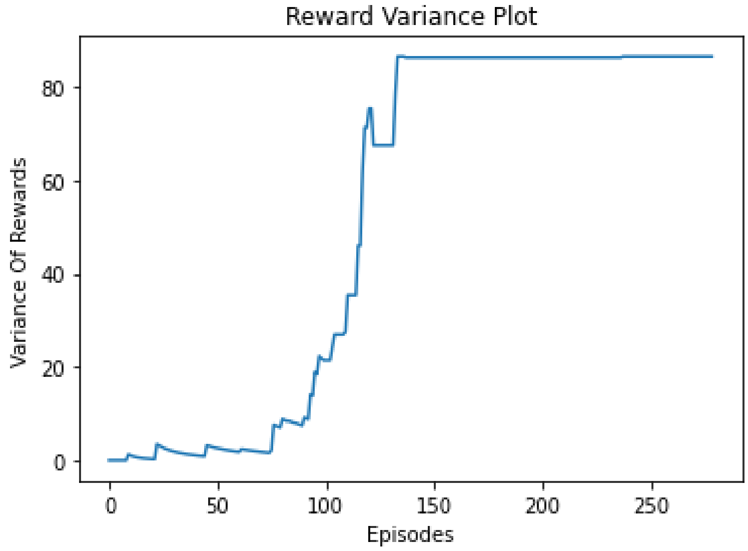




- **Hyper-parameters**
  - Methodology: N-step Return with N = 17
  - Learning Rate: 1e-4
  - Architecture: 128 - 64
  - Episodes: 1000

Results:
Average Episodes to converge = 168.4, Convergence Rate = 25%

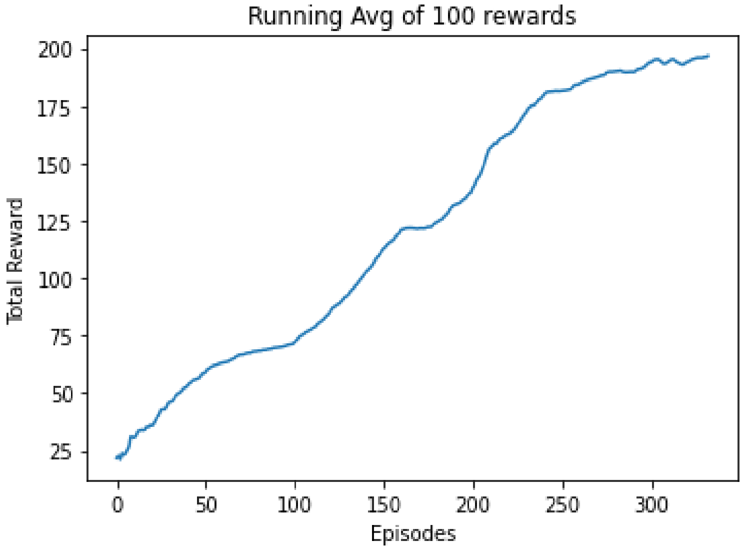


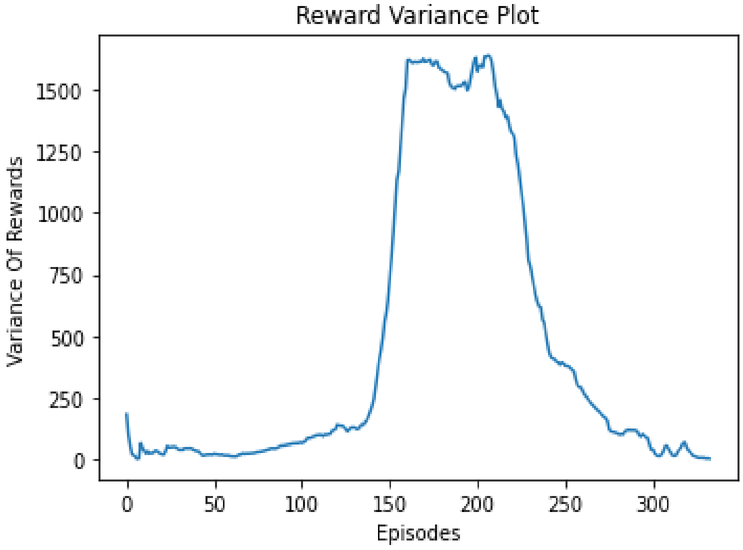


CartPole

- **Hyper-parameters**
  - Methodology: One Step
  - Learning Rate: 1e-4
  - Architecture: 1024 - 512

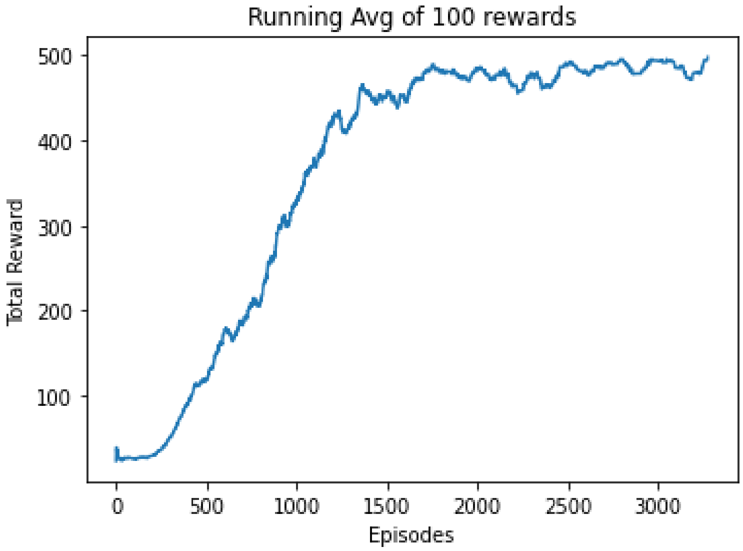
Results:
Average Episodes to converge = 178.6, Convergence Rate = 86%

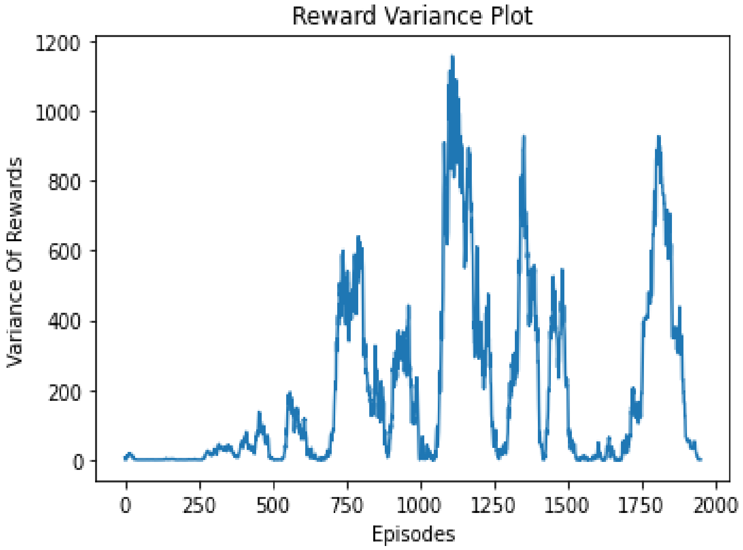




- **Hyper-parameters**
  - Methodology: Full Return
  - Learning Rate: 1e-
  - Architecture: 1024 - 512

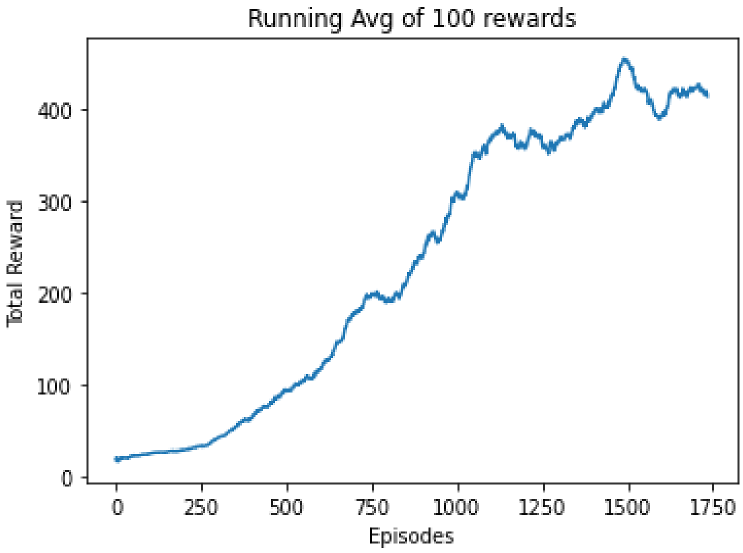
Results:
Average Episodes to converge = 1782.8, Convergence Rate = 80%

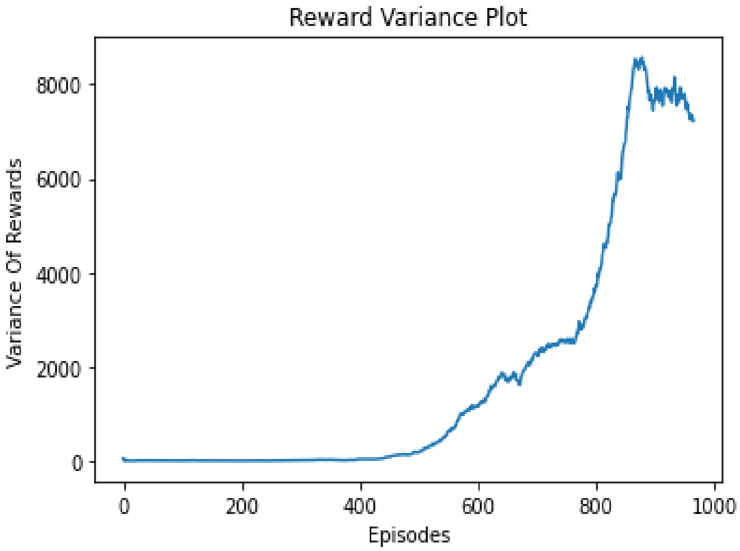




- **Hyper-parameters**
  - Methodology: N-step Return with N = 3
  - Architecture: 1024 - 512
  - Learning Rate: 1e-3

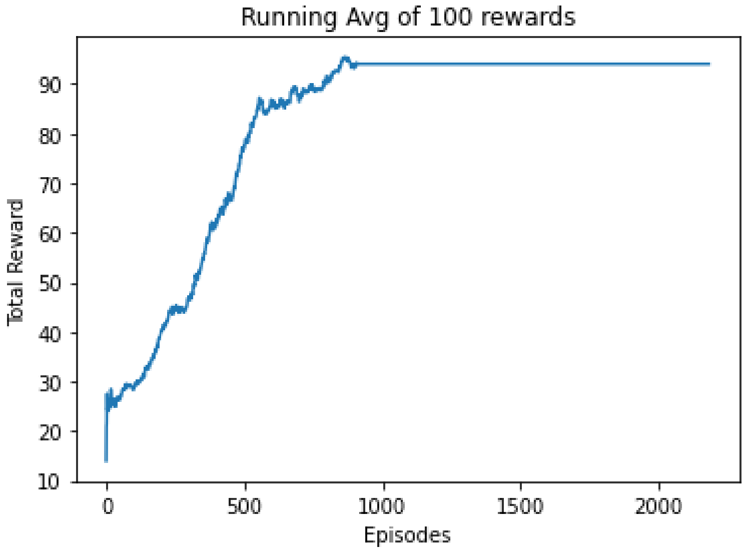
Results:
Average Episodes to converge = 1560.4, Convergence Rate = 60%

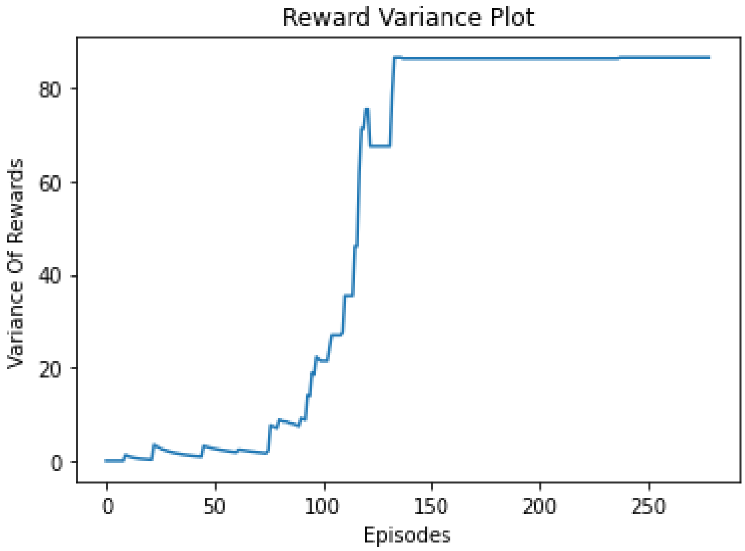




- **Hyper-parameters**
  - Methodology: N-step Return with N = 7
  - Architecture: 128 - 64
  - Learning Rate: 1e-3

Results:
Average Episodes to converge = 2000, Convergence Rate = 30%

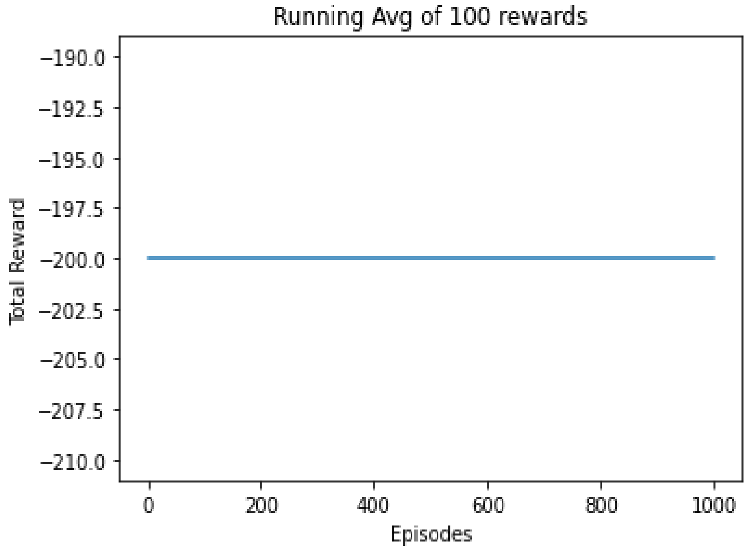


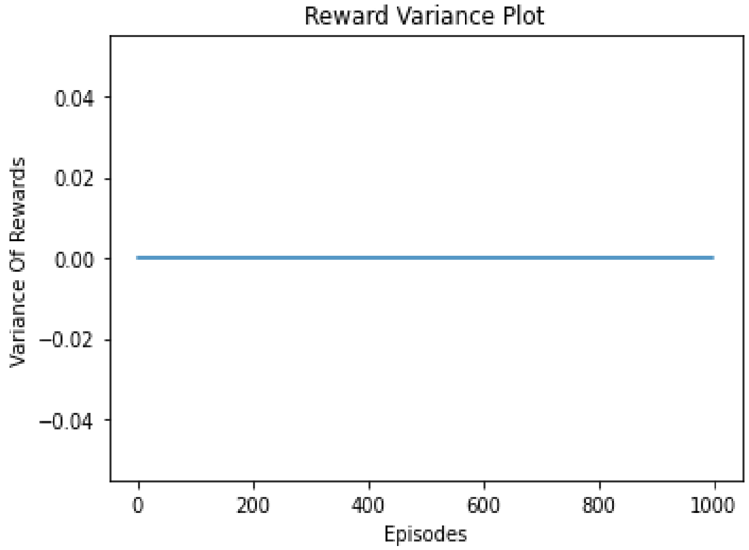


MountainCar

- **Hyper-parameters**
  - Methodology: One Step
  - Learning Rate: 1e-4
  - Architecture: 2048 - 1024

Results:
Average Episodes to converge = 1000, Convergence Rate = 0%

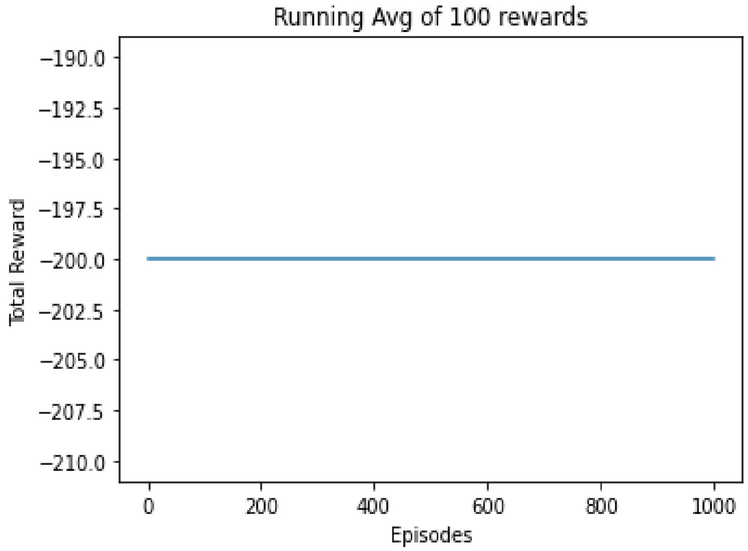


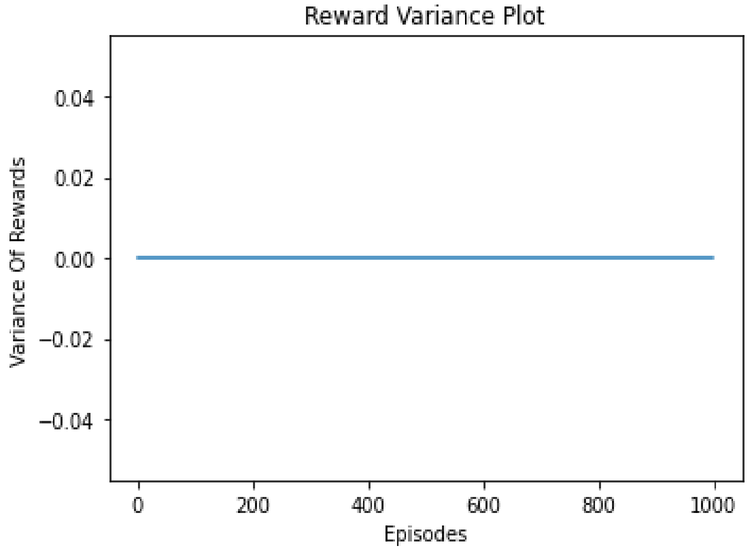


- **Hyper-parameters**
  - Methodology: Full Return
  - Learning Rate: 1e-3
  - Architecture: 2048 - 1024

Results:

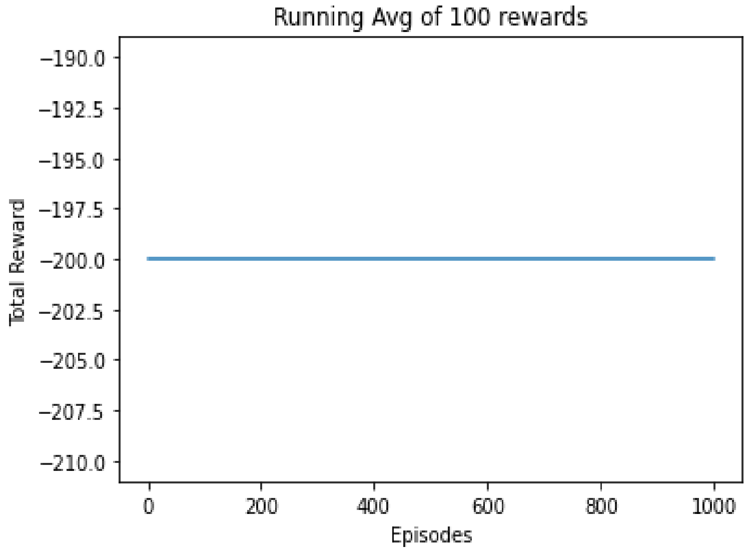
Average Episodes to converge = 1000, Convergence Rate = 0%

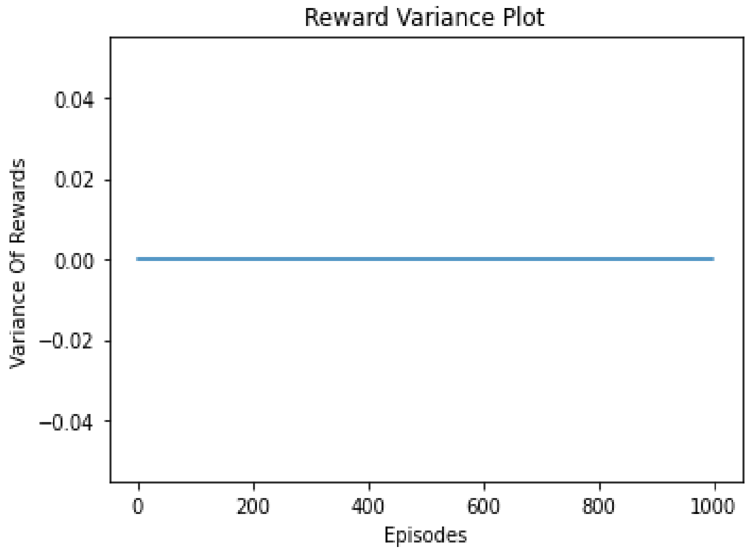




- **Hyper-parameters**
  - Methodology: N-step Return with N = 31
  - Learning Rate: 1e-3
  - Architecture: 2048 - 1024

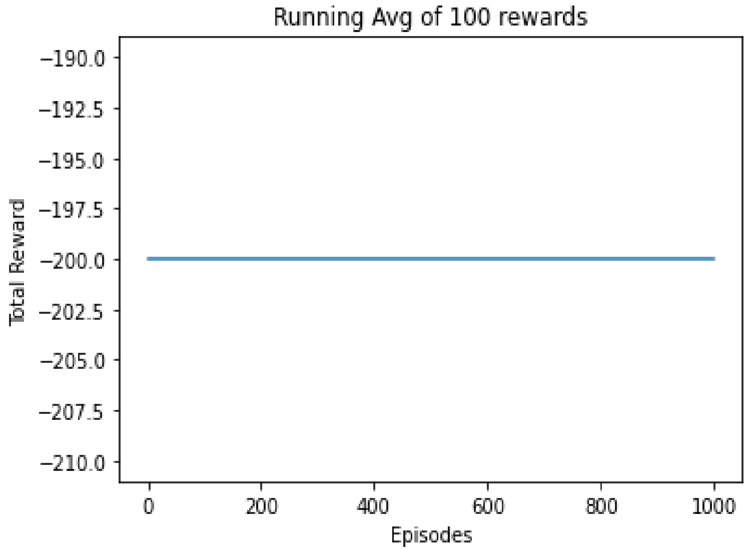
Results:
Average Episodes to converge = 1000, Convergence Rate = 0%

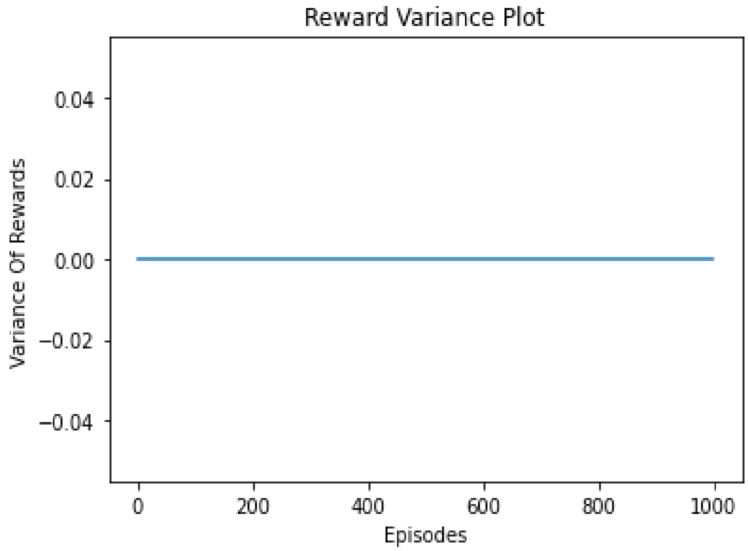




- **Hyper-parameters**
  - Methodology: N-step Return with N = 13
  - Learning Rate: 1e-3
  - Architecture: 2048 - 1024

Results:
Average Episodes to converge = 1000, Convergence Rate = 0%





5 Actor-Critic Inference

- **Acrobot:** The agent was able to learn smoothly in one step methodology, whereas full return and n-step had fluctuations during the learning process.
- However, these fluctuations didn't have any significant impact on the upset. We Observed the variance to increase and decrease for One Step return whereas for the Full Return case the variance seems to be decreasing.
- In the N-Step case we observe that the variance plot seems to be increasing as the episodes increases.
- The variance tends to increases, however on the verge on convergence, it falls back to zero.
- **CartPole:** The agent was relatively easily able to learn using Actor-Critic, compared to other environments. The variance increases with increase in number of episodes.
- The variance seems to be increase during the middle of episodes and decrease during the later episodes.
- **MountainCar:** Even with multiple attempts of hyper-parameter tuning, the agent was unable to solve the environment in all 3 methodologies. Tweaking the reward function could be one possible way to resolve this issue.