Prediction Challenge 3 – Deep Reinforcement Learning

Introduction:

In this challenge, the task is to use the Deep Q Network (DQN) algorithm to replace the human element in the CartPole-v0 environment in OpenAI Gym. The goal is to train a model that can balance the pole on the trolley for 200 steps for 20 consecutive episodes, with the training occurring in the least possible amount of episodes.

Model Explanation:

The model architecture consists of an input layer that takes in one observation vector, and the number of observations in that vector. The output layer is a fully connected layer with the number of units equal to the number of actions in the action space and it includes one hidden layer with 16 units, respectively.

The model is compiled using the Adam optimizer with a learning rate of 1e-3 and mean absolute error (mae) as the metric. The model is then trained for 10000 steps using the fit method of the DQN Agent.

```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 flatten_2 (Flatten)         (None, 4)                 0

 dense_7 (Dense)             (None, 16)                80

 dense_8 (Dense)             (None, 2)                 34

=================================================================
Total params: 114
Trainable params: 114
Non-trainable params: 0
_________________________________________________________________
None
```

Fig:1

To evaluate the performance of the model, testing is conducted for 20 episodes, with the goal of achieving a reward of 200 in each episode. The model was able to achieve this goal, as shown in the test results presented in the submission.

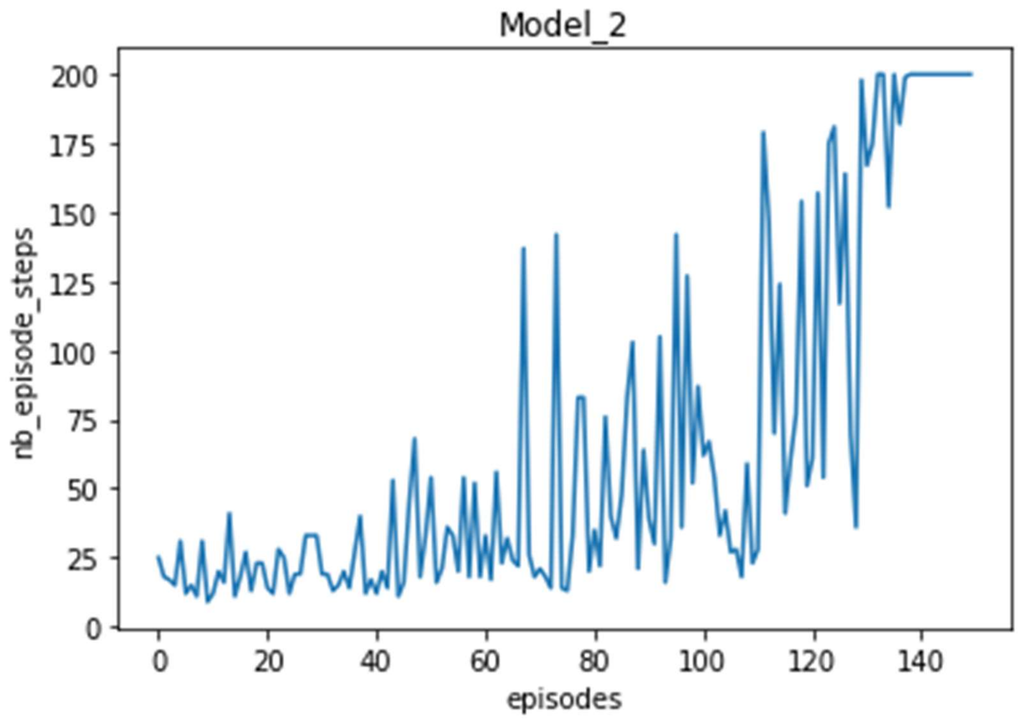

Fig:2

```
for i in range(0,10):
    dqn_1.test(env, nb_episodes=20, visualize=False)
    print('\n')
```

```
Testing for 20 episodes ...
Episode 1: reward: 200.000, steps: 200
Episode 2: reward: 200.000, steps: 200
Episode 3: reward: 200.000, steps: 200
Episode 4: reward: 200.000, steps: 200
Episode 5: reward: 200.000, steps: 200
Episode 6: reward: 200.000, steps: 200
Episode 7: reward: 200.000, steps: 200
Episode 8: reward: 200.000, steps: 200
Episode 9: reward: 200.000, steps: 200
Episode 10: reward: 200.000, steps: 200
Episode 11: reward: 200.000, steps: 200
Episode 12: reward: 200.000, steps: 200
Episode 13: reward: 200.000, steps: 200
Episode 14: reward: 200.000, steps: 200
Episode 15: reward: 200.000, steps: 200
Episode 16: reward: 200.000, steps: 200
Episode 17: reward: 200.000, steps: 200
Episode 18: reward: 200.000, steps: 200
Episode 19: reward: 200.000, steps: 200
Episode 20: reward: 200.000, steps: 200
```

Fig:3

Critical Appraisal of the model:

With the trial-and-error method only combination of hyperparameters and parameters was chosen for the models. I tried with multiple parameters in which, I observed that the few layers in the model giving the better rewards and model is performance is good. Through this process, I found out that adding layers did not necessarily improve the results. To evaluate the model's performance, I did test on the similar model multiple times to check for the consecutive steps and the rewards, for this model.

Conclusion:

In conclusion, the model was trained effectively and achieved the desired results for the task. The experimentation process allowed for the identification of the optimal model configuration, while the testing provided a reliable evaluation of the model's performance.

Word Count: 318

Code Link:

References:

1. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning (Vol. 1). MIT press.
2. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436-444.
3. Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., ... & Petersen, S. (2015). Human-level control through deep reinforcement learning. Nature, 518(7540), 529-533.
4. https://github.com/chambai/Deep_Learning_Course/blob/main/Week%203%20Deep%20RL%201/CartPoleQLearning.ipynb