CS 5392 Spring 2023 Assignment-1 ExampleRun-3

To run the examples:-

Commands to Visualize logs

To visualize graphs representation of software metrics, we have saved our logs from training using tensorbaord.

go to the folder where logs and models are saved.In the same folder where we run our tutorial_2.py

i.e F:\RLL\PythonAPI\examples

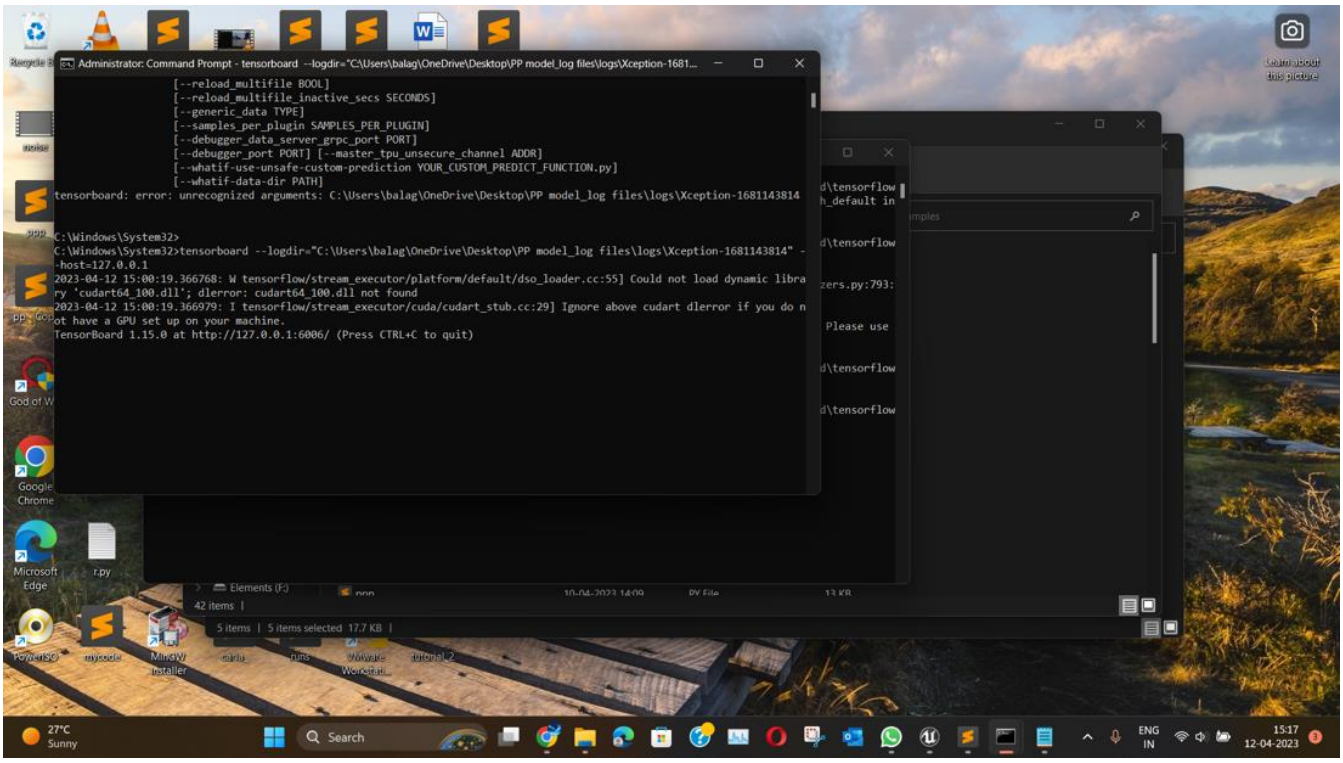
Now go to logs folder, copy the full path of the log folder and then run the command in cmd

$ tensorboard --logdir="Your saved log_dir path" --host=127.0.0.1

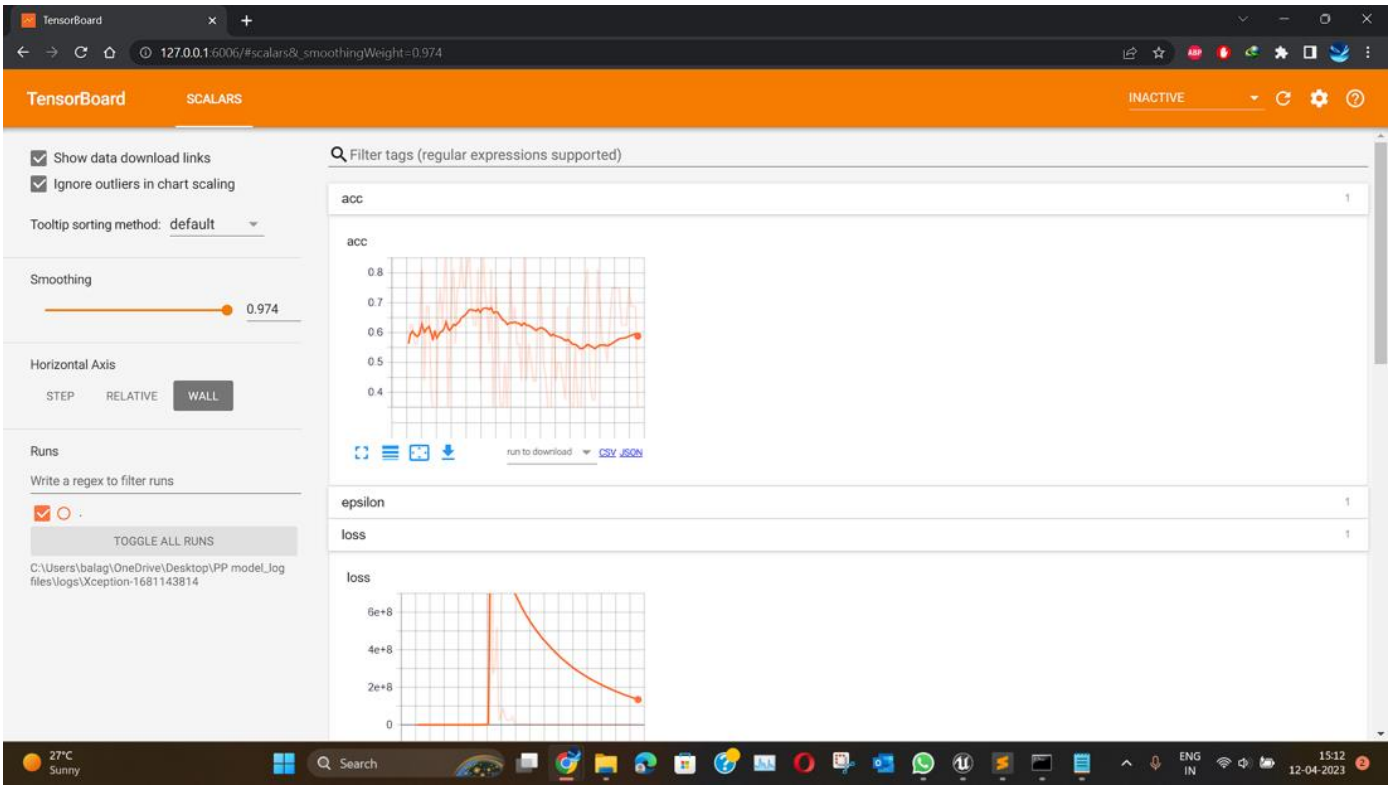
Example:-

$ tensorboard --logdir="F:\RLL\PythonAPI\examples\logs\Xception-1681346435" --host=127.0.0.1

once you run the command, you will get . TensorBoard 1.15.0 at http://127.0.0.1:6006/ copy the http://127.0.0.1:6006/ and paste it in the any of the browser. we used chrome and click on enter. you will get different type of graphs from the log file.



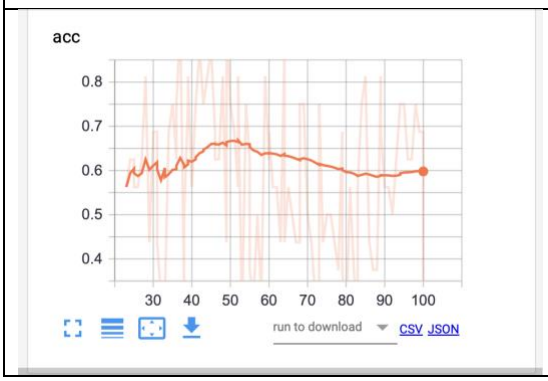
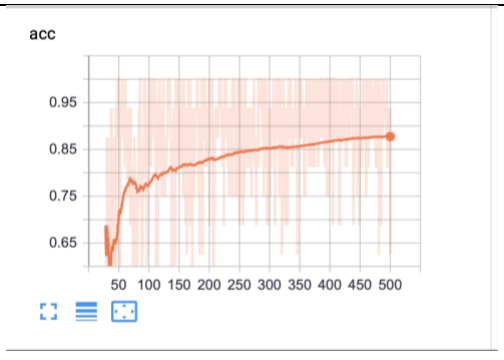
In the tensor board you will see different metrics like accuracy, loss, epsilon etc.

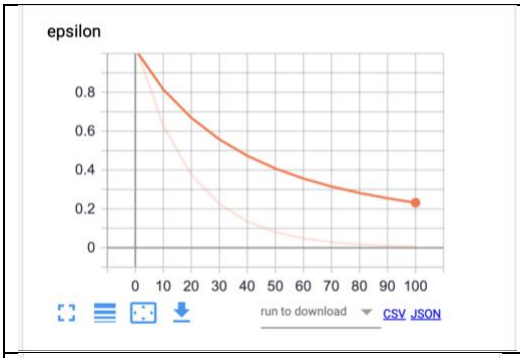


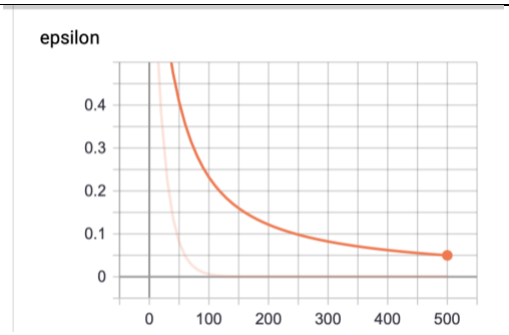
Test Cases: -

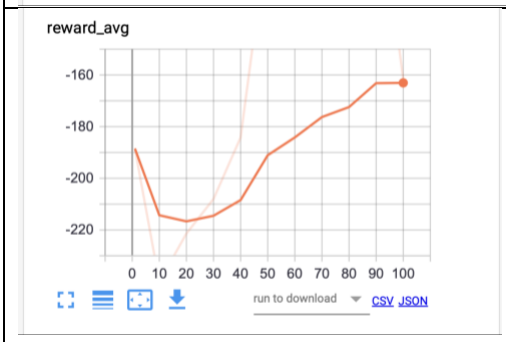
Here are comparisons of the test cases of 100 and 500 episodes. We have attached the tensor board images for reference in tensorboard folder **that has pdfs of Xception , CNN and 5 layered CNN for 100 episodes and 500 episodes along with reward change trained for 100 episodes**

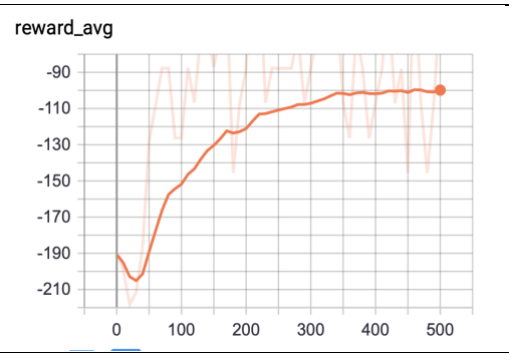
Xception

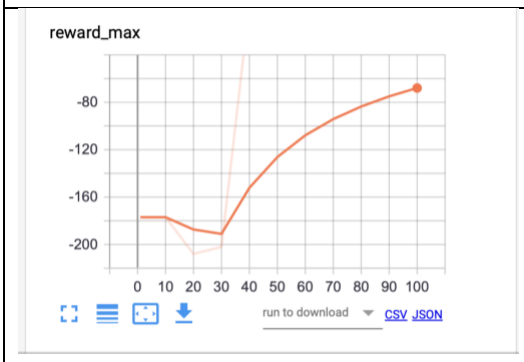
100 Episodes:-	**500 Episodes:-**
	

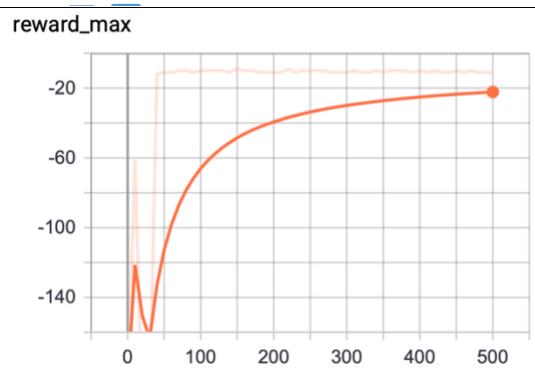

epsilon
0.8
0.6
0.4
0.2
0
0 10 20 30 40 50 60 70 80 90 100
run to download
CSV JSON

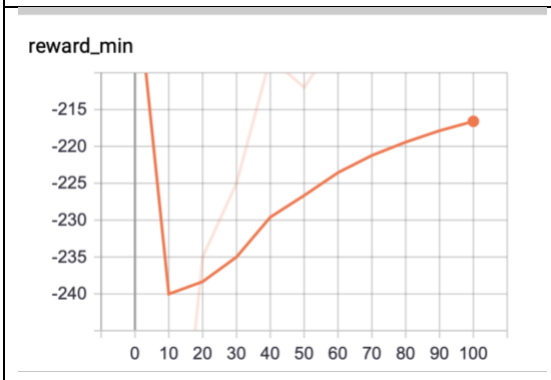

epsilon
0.4
0.3
0.2
0.1
0
0 100 200 300 400 500

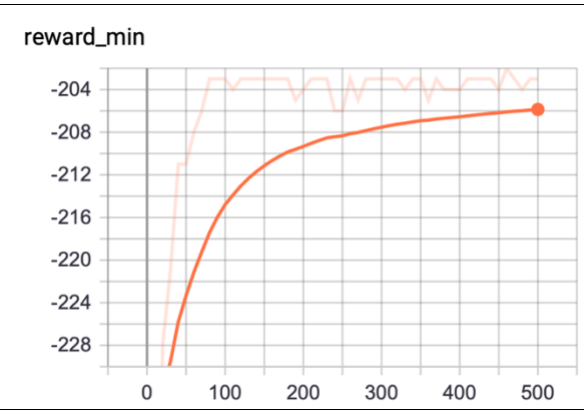

reward_avg
-160
-180
-200
-220
0 10 20 30 40 50 60 70 80 90 100
run to download
CSV JSON


reward_avg
-90
-110
-130
-150
-170
-190
-210
0 100 200 300 400 500

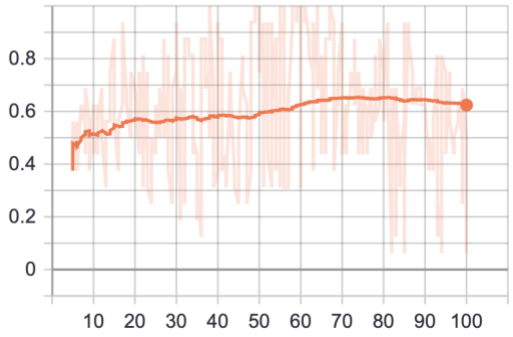
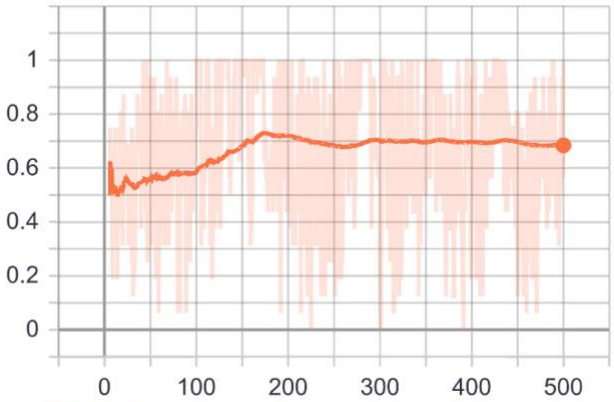
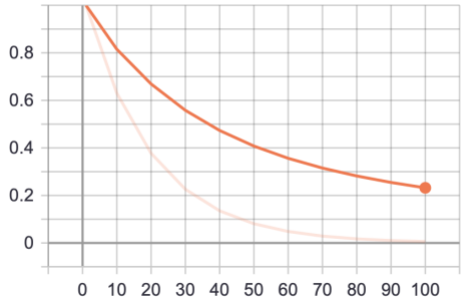
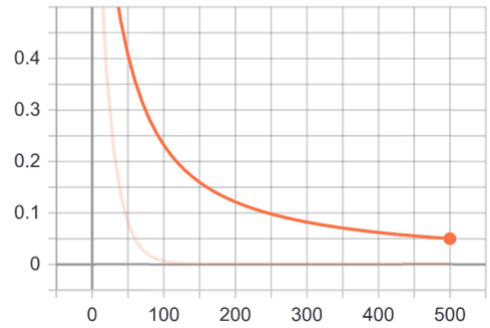
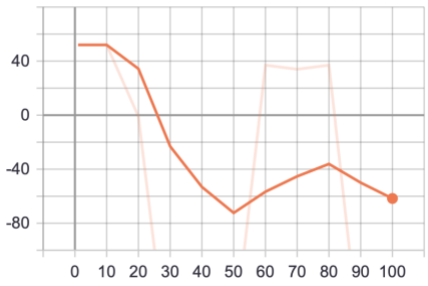
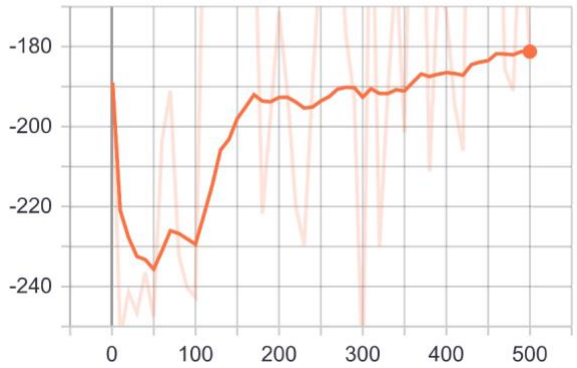
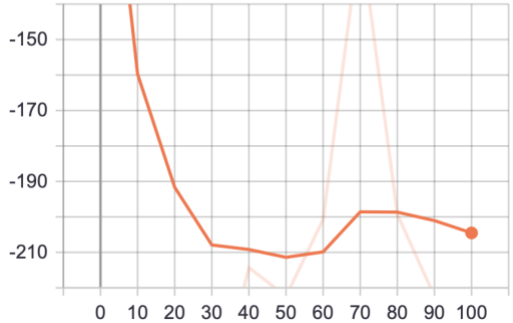
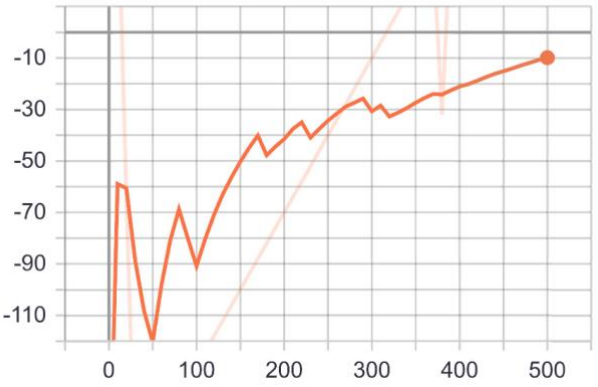

reward_max
-80
-120
-160
-200
0 10 20 30 40 50 60 70 80 90 100
run to download
CSV JSON

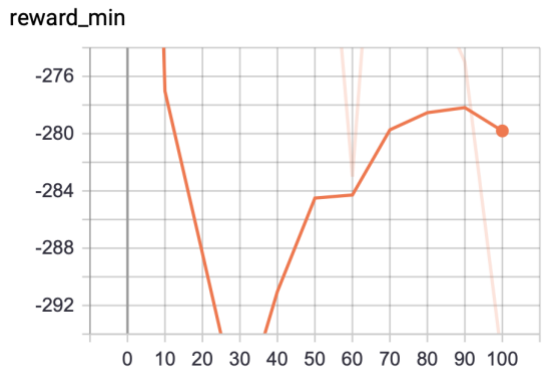

reward_max
-20
-60
-100
-140
0 100 200 300 400 500

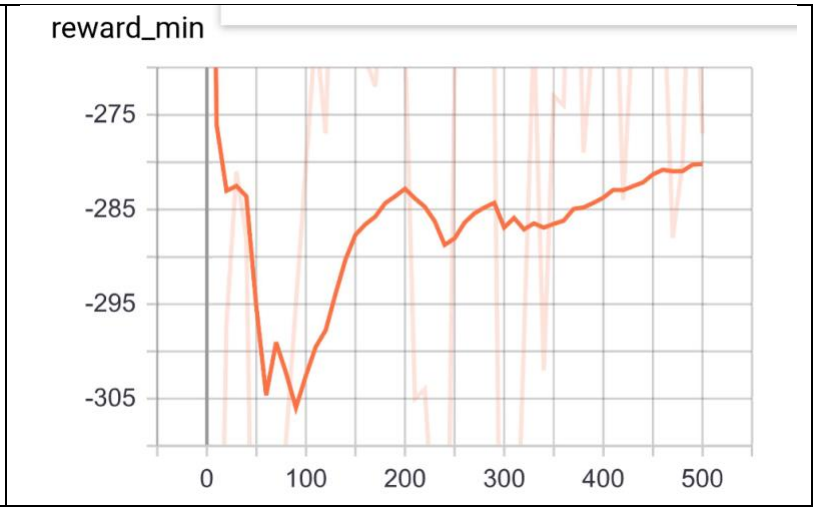

reward_min
-215
-220
-225
-230
-235
-240
0 10 20 30 40 50 60 70 80 90 100


reward_min
-204
-208
-212
-216
-220
-224
-228
0 100 200 300 400 500

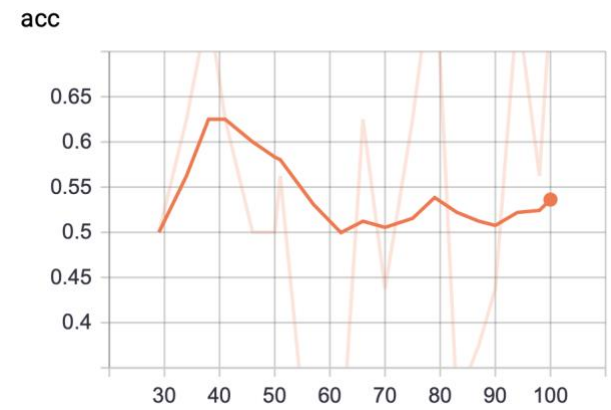
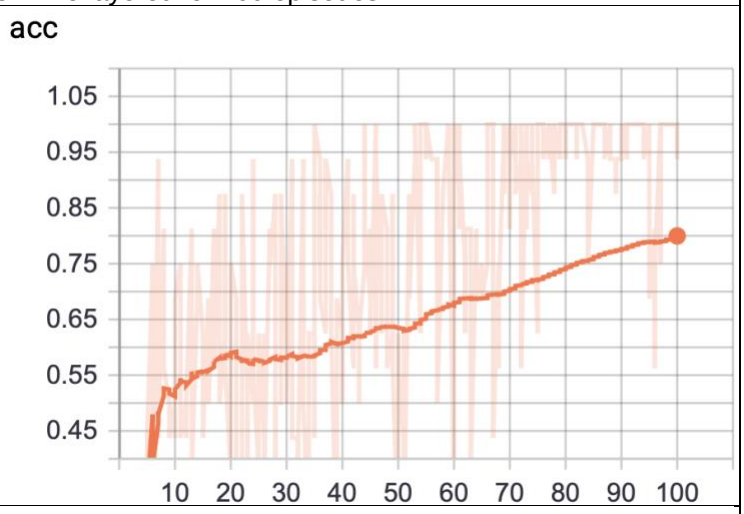
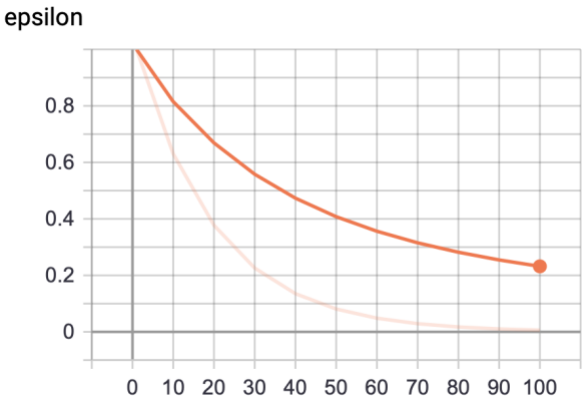
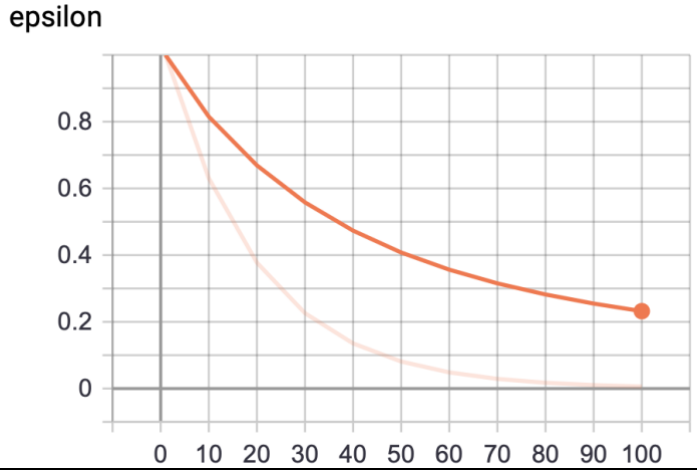
CNN 5 Layered:-

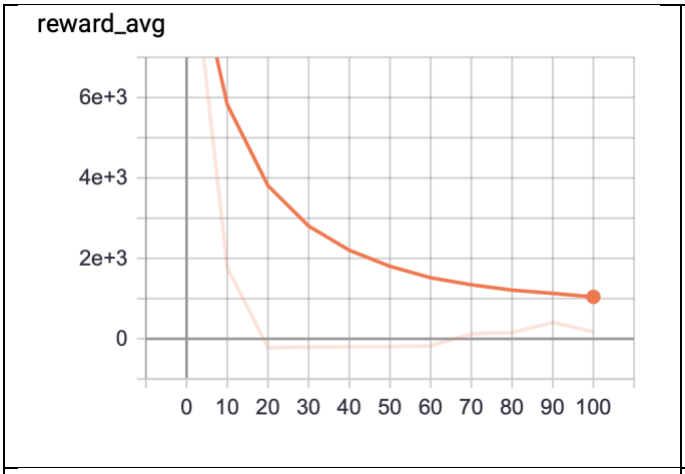
100 Episodes:	500 Episodes:
acc 	acc 
epsilon 	epsilon 
reward_max 	reward_avg 
reward_avg 	reward_max 

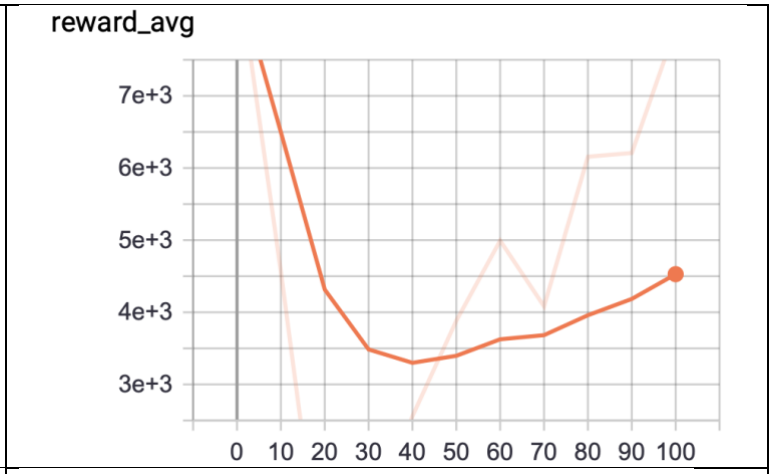


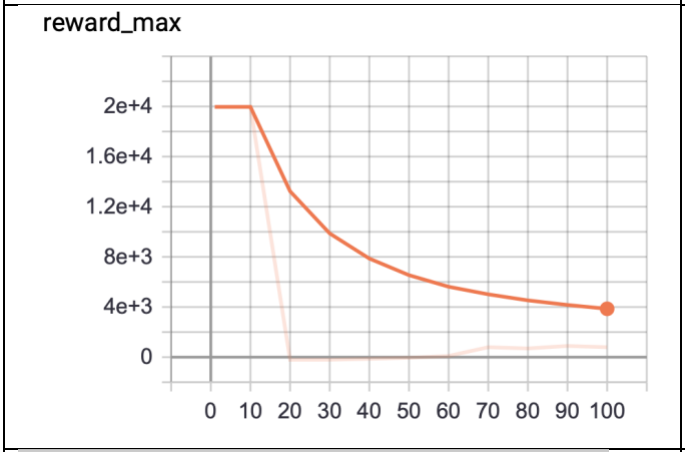


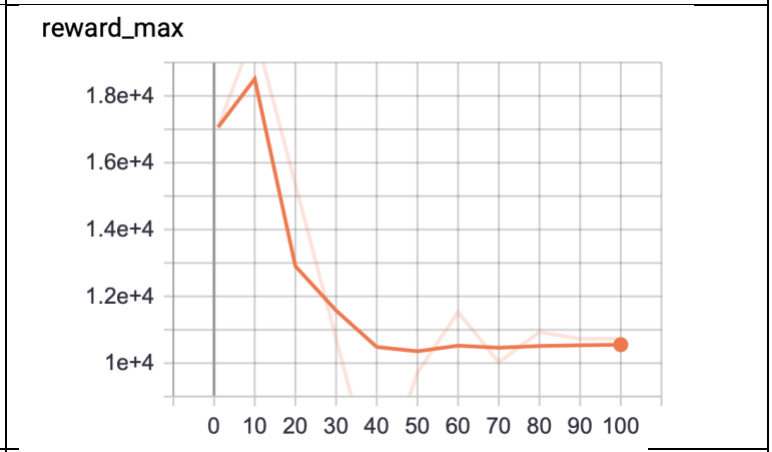
Reward Change:-

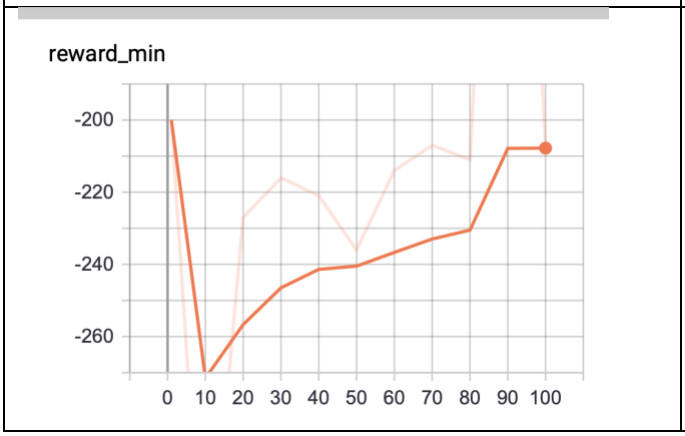
Xception for 100 episodes	CNN- 5 layered for 100 episodes
	
	

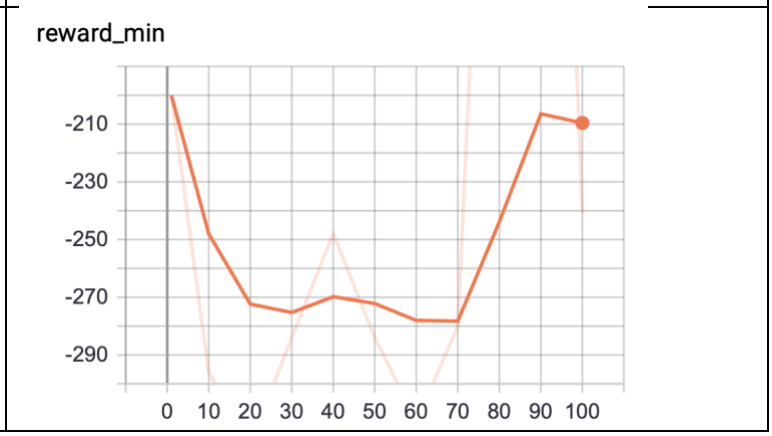












The evaluation results show that as the number of episodes increases from 100 to 500, there is an improvement in the accuracy of the RL agent indicating that the agent is learning and improving its decision-making ability over time. With the decrease in epsilon, the agent is gradually shifting from exploration (trying new actions) to exploitation (choosing actions with the highest Q-values) as it gains more experience from rewards, the agent is learning to make better decisions and achieving higher rewards over time, but there is still some variability in the rewards obtained. Further analysis and fine-tuning may be needed to optimize the performance of the RL agent.

Comparing the models Xception and CNN , Xception has higher accuracy but CNN is taking less time to train compared to Xception. Moreover, the reward has little difference in both models. But to train our model Xception is the best model.