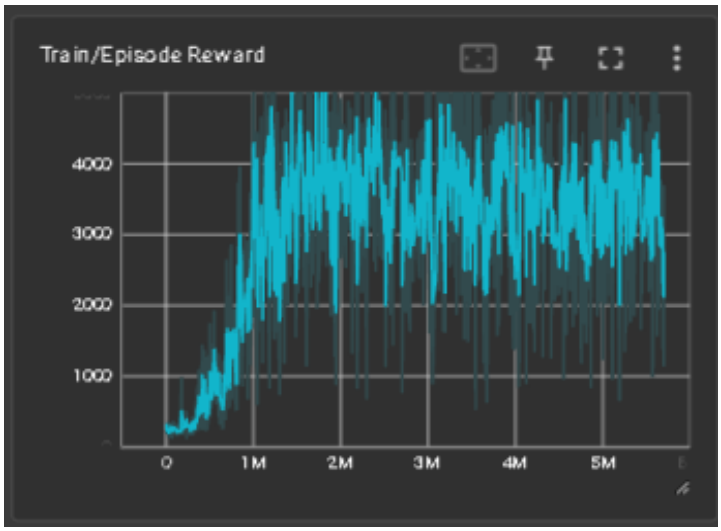
RL Lab2-DQN

學號:313554044 姓名:黃梓誠

Screenshot of Tensorboard training curve and testing results on DQN

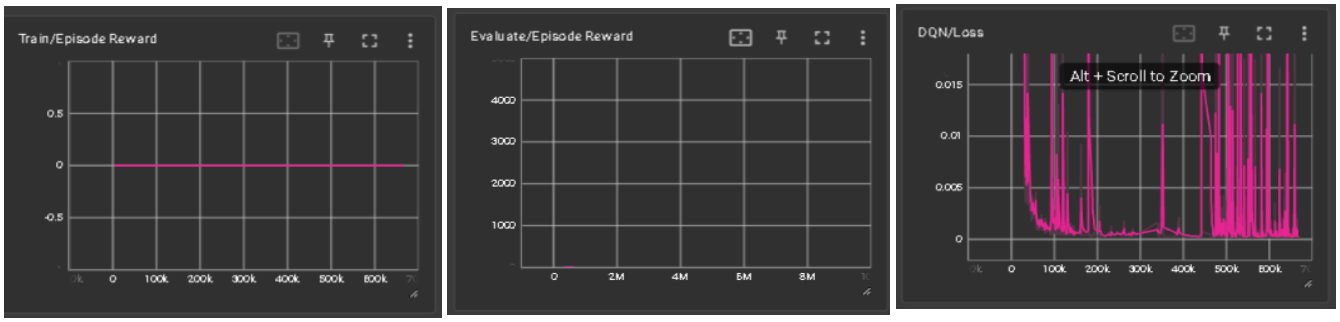
Training curve:



Testing results (5 games):

```
Evaluating...
episode 1 reward: 4230.0
episode 2 reward: 3840.0
episode 3 reward: 4470.0
episode 4 reward: 4210.0
episode 5 reward: 4120.0
average score: 4174.0
```

Screenshot of Tensorboard training curve and testing results on Enduro-v5 using DQN (10%).



我在最後一兩天測試 Enduro-v5 DQN，但發現他無法取得 reward。以下是我分析的幾點原因與改進方向

1. Enduro-v5 中 reward 相當 sparse：
Enduro 是一款獎勵稀少的賽車遊戲。與 Ms. Pacman 頻繁獲得獎勵（吃點點數）不同，在 Enduro 中，agent 僅收到過去車輛的獎勵。這意味著 agent 可能會經歷很長的 sequence 而不會收到任何獎勵，特別是在訓練的早期，它還不擅長超越其他車輛時。

可能的解決方案：確保在訓練的早期有足夠大的 replay buffer 和足夠的 exploration（使用更高的 epsilon 值），以使 agent 有更多機會學習到罕見的獎勵。

2. **Long Episode Lengths**:

```
[460282/500000]  episode: 139  episode reward: 0.0  episode len: 3305  epsilon: 0.60362650001305
[463584/500000]  episode: 140  episode reward: 0.0  episode len: 3303  epsilon: 0.6006538000131478
[466897/500000]  episode: 141  episode reward: 0.0  episode len: 3314  epsilon: 0.597671200013246
[470223/500000]  episode: 142  episode reward: 0.0  episode len: 3327  epsilon: 0.5946769000133446
[473536/500000]  episode: 143  episode reward: 0.0  episode len: 3314  epsilon: 0.5916943000134428
```

在 log 中，**Episode Length** 相當大（大約 3300），但 reward 為 0。這可能代表 agent 在遊戲中沒有取得進展，可能只是在沒有超車的情況下駕駛。或許 agent 沒有學習到如何改進。

可能的解決方案：可以考慮調整獎勵結構
（例如，對停留時間過長而沒有超車的情況給予輕微的負獎勵），以鼓勵 agent 採取更快地超車的行動。

3.動作空間複雜度：
與 Ms. Pacman 相比，Enduro 的動作空間相對較小。這使得 agent 更容易探索所有可能的操作。然而，DQN 可能會在環境具有時間依賴性時陷入劣勢，

意思是 agent 需要學習做出一系列動作才能超越汽車，而使用標準 DQN 可能需要更長的時間來學習。

可能的解決方案：使用 Double DQN (DDQN)或 Dueling DQN，等幫助 agent 做出更明智決策的能力。

4.Enduro 中的獎勵訊號太稀疏，DQN 可能很難學習。

可能的解決方案：嘗試獎勵標準化或獎勵裁剪，以確保獎勵在 DQN 可以處理的範圍內。

◆ Screenshot of Tensorboard training curve and testing results on DDQN, and discuss the difference between DQN and DDQN (3%).

觀念:

$$Y_t^{Q} = r_{t+1} + \gamma \max_{a} Q(S_{t+1}, a|\theta^{-})$$

$$Y_t^{DoubleQ} = r_{t+1} + \gamma Q\left(S_{t+1}, \operatorname{argmax}_{a} Q(S_{t+1}, a|\theta)|\theta^{-}\right)$$

1. DQN suffers from over-estimation:

- DQN 使用同一個 Q network 來選擇和評估最佳動作，這會導致高估 Q 值，並進而影響決策品質。

2. Behavior and Target Network (行為網絡與目標網絡)**:

- DDQN 使用兩種 networks：

1. behavior network： 用於根據當前策略選擇動作

2. target network： 用於提供更穩定的 Q 值估計。目標網絡更新的頻率較低，以保持穩定性。

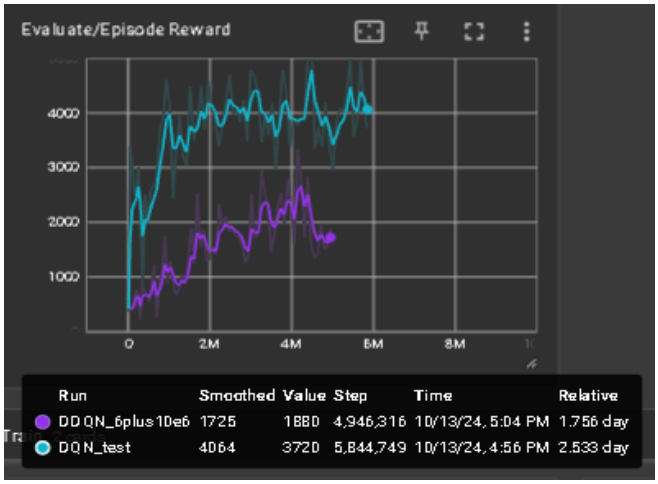
3. Reduce the over-estimation problem:

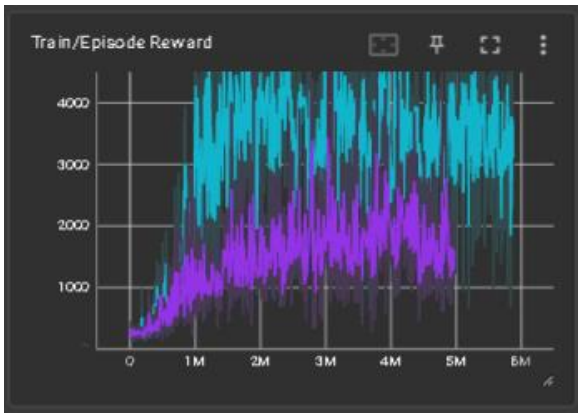
在 DDQN 中，行為網絡選擇最佳動作，而目標網絡則評估該動作的 Q 值，這樣可以降低 Q 值被高估的可能性，並改善學習效果。

實作:

```
with torch.no_grad():
    ################ ver1 ##############################
    # calculate q_next of behavior nn,
    q_next_behavior = self.behavior_net(next_state)

    # find argmax action of brhavior nn      # q_next_behavior:形狀為 [batch_si
    argmax_action_behavior = q_next_behavior.argmax(dim=1).unsqueeze(1)  # 在

    # use argmax_action_behavior on target net to get q_next
    q_next = self.target_net(next_state).gather(1,argmax_action_behavior )

    # if episode terminates at next_state, then q_target = reward
    q_target = reward + self.gamma * q_next * (1 - done)
```

實作中的結果意外發現 DQN 在一開始的訓練反而比較好，因為設備緣故我無法訓練完 1e8 epoch，我認為 DDQN 較差可能是訓練時長不夠所以沒有發揮效果。



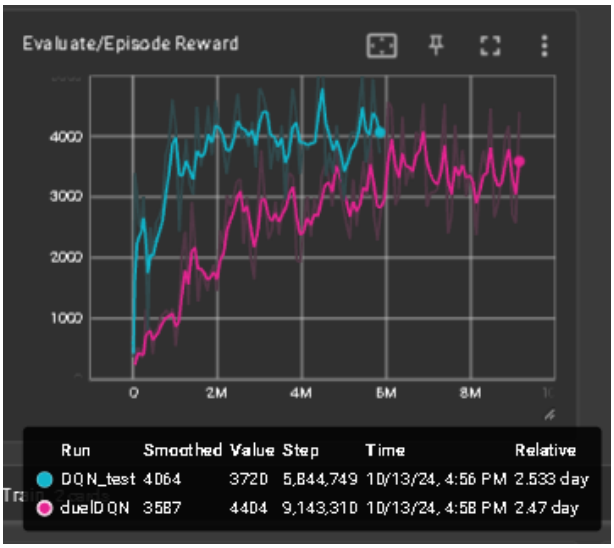


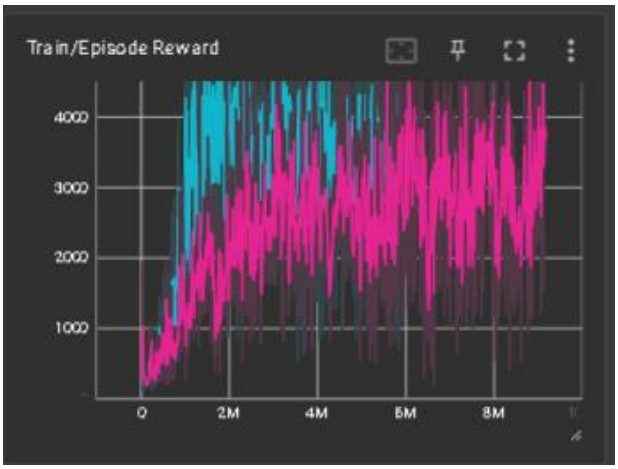
◆ Screenshot of Tensorboard training curve and testing results on Dueling DQN,and discuss the difference between DQN and Dueling DQN

觀念:

1. 在 Dueling DQN 中，Q-value 被分成兩個獨立的分支：一個是 Advantage function，另一個是 Value function。Advantage function 代表在給定狀態下每個動作的相對重要性，而 Value function 則表示該狀態的整體價值。
2. 由於 A 值的數值範圍相對較小，對模型更新更加敏感，使得模型更容易考慮與其他動作相關的相對變化。
3. 另外，Dueling DQN 包含一個機制來限制 Advantage function 的值，從而避免 Q-value 中的誤判。這是用減去所有動作的 Advantage 值的平均來實現的。這確保了 Q-value 能夠更好地被 normal，並更有效地反映動作之間的**相對差異**，避免 Advantage 值可以通過加上一個常數，而不改變 Q-value 的問題。

實作:






```python
#Q value
def forward(self, x):
    # x = torch.unsqueeze(x, dim=1)
    x = x.float() / 255.
    x = self.cnn(x)
    x = torch.flatten(x, start_dim=1)
    advantage = self.advantage(x)
    value = self.value(x)
    return value + advantage - torch.mean(advantage,dim =1,keepdim=True)
```

可以看到 duel 是穩健上升的而 DQN 雖然比較快上升但接著就一直卡了，如果時間尺度拉長或許可以有更好的效果

◆ **Training curve comparison (DQN vs. DDQN vs. Dueling DQN)**