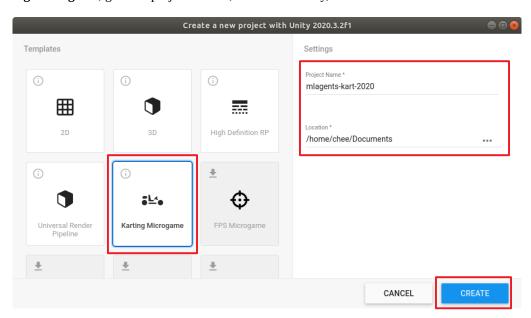
# Creating a New Environment For Karting Micrograme 3.0.0 + ML-Agents <a href="https://learn.unitv.com/tutorial/karting-mod-smart-karts-training-guide#5ec832feedbc2a31b5a891f4">https://learn.unitv.com/tutorial/karting-mod-smart-karts-training-guide#5ec832feedbc2a31b5a891f4</a>

This demo was done on **Unity 2020.3.2f1**. Karting Micrograme 3.0.0 is only compatible with Unity 2020. Also, this version of Karting Micrograme does not require Karting Academy.

- 1) Prerequisites:
  - 1.1 Python environment with MLAgents 0.16.1 Python package installed
    - 1.1.1 Create a new environment
      - conda create --name mlkart20 tensorflow
        conda activate mlkart20
    - 1.2.2 Install MLAgents Python packages and check if it was installed
      - pip install -Iv mlagents==0.16.1
        mlagents-learn --help
  - 1.2 Unity Hub 2.4.0 or later (Download/Install Unity Hub from: https://unity3d.com/get-unity/download)
  - 1.3 Unity 2020.3 Editor (Install through **Unity Hub** > **Installs**)
- 2) Create a new Unity project using Unity 2020 Editor



- 3) Unity Editor 2020.3 has latest release of Karting Micrograme; hence, download it by clicking on
- 4) Select **Karting Microgame**, give the project a name, select a directory, and click **Create**



- 5) (Optional) To be more familiar with Karting micrograme such as the following, go through the Tutorials
  - Moving/adding objects (prefabs)
  - Change camera view
  - Move between Scene and Game mode
  - Resize objects
  - Change agent's parameters (speed)
  - Pan the environment

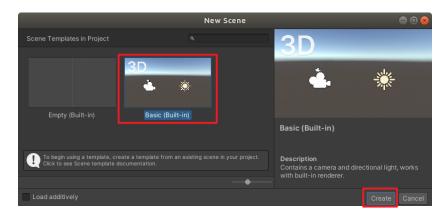


Quick explanations for some useful shortcuts in Scene Mode

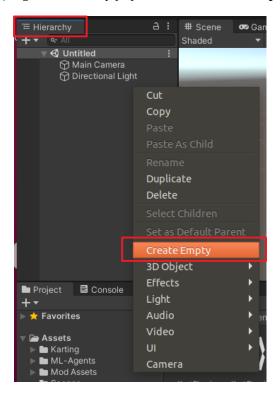
- Hand Tool: To pan around the scene
- Move Tool: Move selected objects in their x, y, z axes by dragging the axis in desired direction
- Rotation Tool: Rotate selected objects around the x, y, z axes by dragging on the rotation outline
- Scale Tool: Scale selected objects along x, y, z axes
- Move/Rotate/Scale Tool : All three
- **Grid Snapping Toggle**: Snaps objects together when using Move, Scale, Rotation tools (set Tool Handle Rotation to Global)



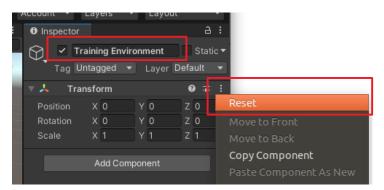
- · Right-click, hold and pan mouse to rotate current view
- While holding right button on mouse, press Q, W, E, A, S, D to move current view up, forward, down, left, down, right, respectively
- 6) Create a new scene by selecting **File** > **New Scene** > **3D Basic** and click on **Create**



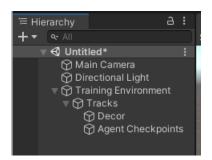
7) In **Hierarchy** (window on the left), right-click an empty spot and select **Create Empty** 



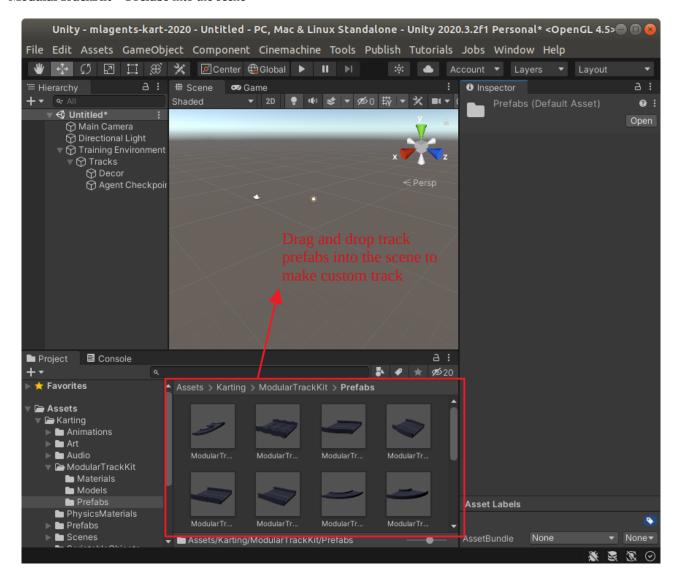
- 8) Rename it **Training Environment** in the **Inspector** (window on the right)
- 9) Reset position of **Training Environment** to origin by clicking on the three dots in **Transform** and choosing **Reset**



- 10) Right-click **Training Environment** and select **Create Empty**. Rename it **Tracks**
- 11) Right-click **Tracks** and select **Create Empty** for: **Decor** and **Agent Checkpoints**. In the end, it should be the following

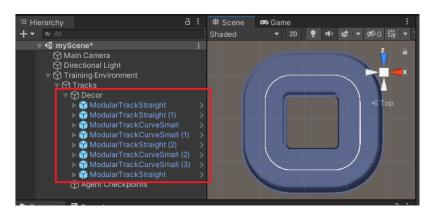


12) Start building a track by dragging and dropping the track prefabs from **Project** > **Assets** > **Karting** > **ModularTrackKit** > **Prefabs** into the scene

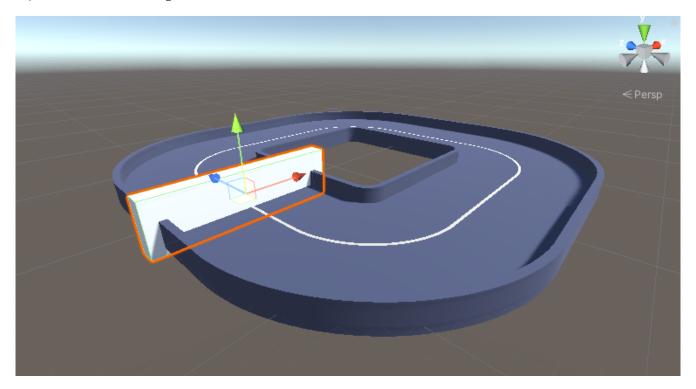


## Tips:

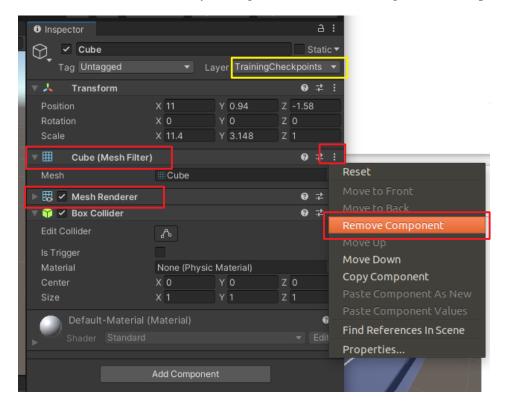
- Set y-value to 0 for all tracks in **Transform** > **Position** > **Y** (y-axis is horizontal placement in the scene)
- Use Shift key to select multiple objects
- Duplicate one or more objects by pressing CTRL+D (or CMD+D)
- 13) Once the track is built, place them all in **Decor**



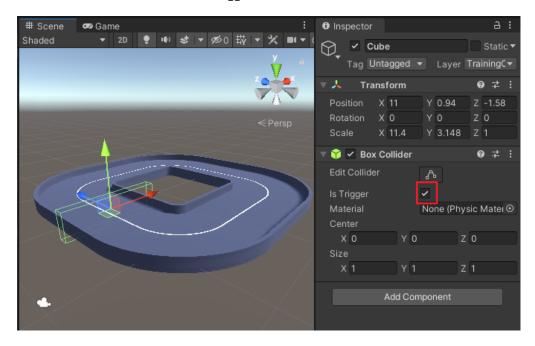
- 14) Right-click **Agent Checkpoints** and select **3D Object** > **Cube** (this is the box collider the agents use as checkpoints)
- 15) Resize box collider using **Scale Tool** to fit the width of the track as shown below



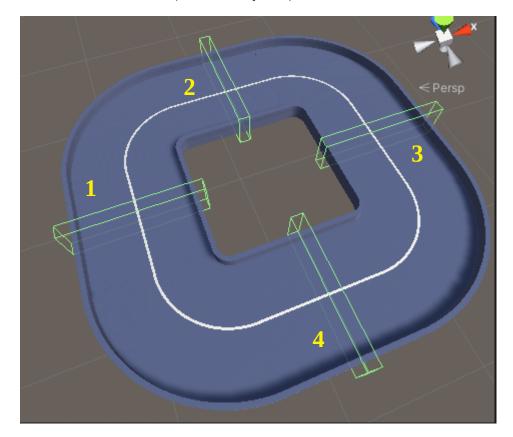
- 16) While cube is selected, select **TrainingCheckpoints** for the **Layer** in the **Inspector** (in yellow box shown below)
- 17) Remove the **Mesh Filter** and **Mesh Renderer** by clicking on three dots and selecting **Remove Component**



## 18) In **Inspector** > **Box Collider**, check the box **Is Trigger**

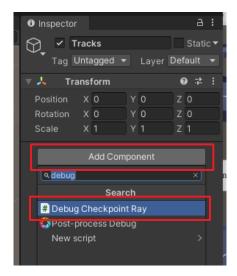


19) Duplicate the the box collider a few times (I used 4 for my track)

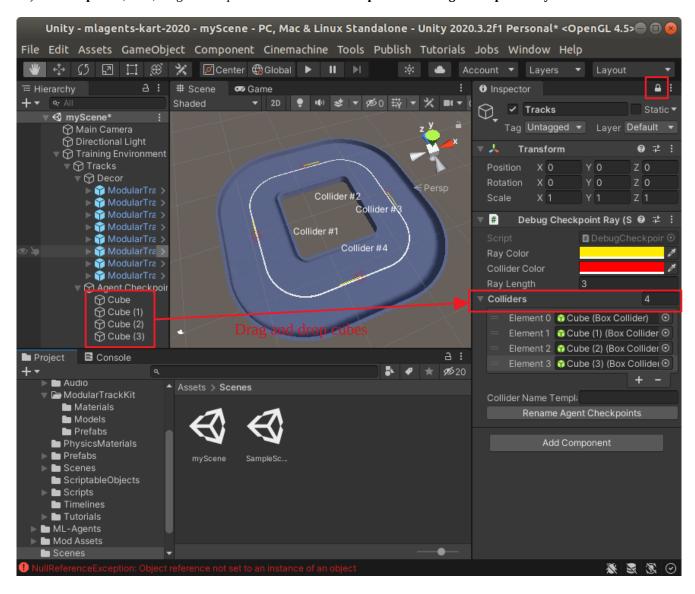


20) Check the placement of box colliders are in order (shown above, this matters since the kart agents use this order to train)

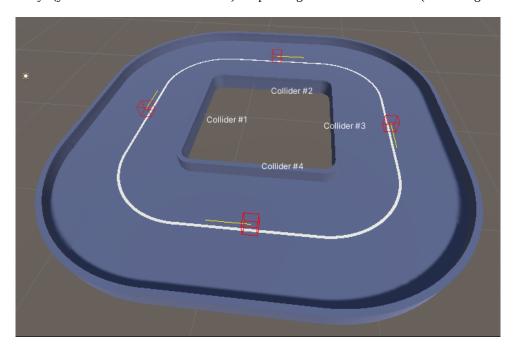
21) Click on Tracks and add a new component Debug Checkpoint Ray in the Inspector



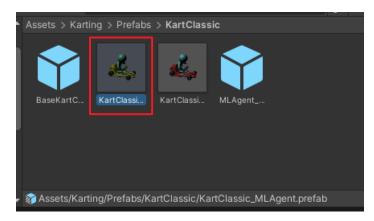
22) Lock **Inspector**; then, drag and drop all box colliders into **Inspector** > **Debug Checkpoint Ray** > **Collider** 



23) Check that the rays (yellow lines out of the red boxes) are pointing in direction of tracks (Don't forget to save)

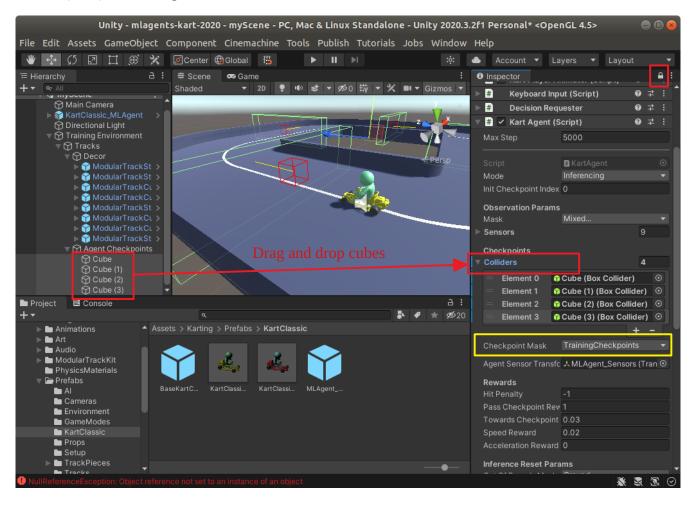


24) Grab a **KartClassic\_MLAgent** prefab from **Project** > **Assets** > **Karting** > **Prefabs** > **KartClassic** onto the track (make sure the kart is just above the track and not in or below it)

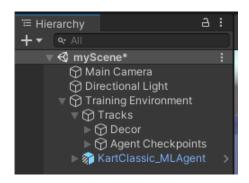




25) Lock the inspector for **KartClassic\_MLAgent** (don't forget to unlock from previous steps); then, drag and drop box colliders (cubes) into **Kart Agent** > **Colliders** 



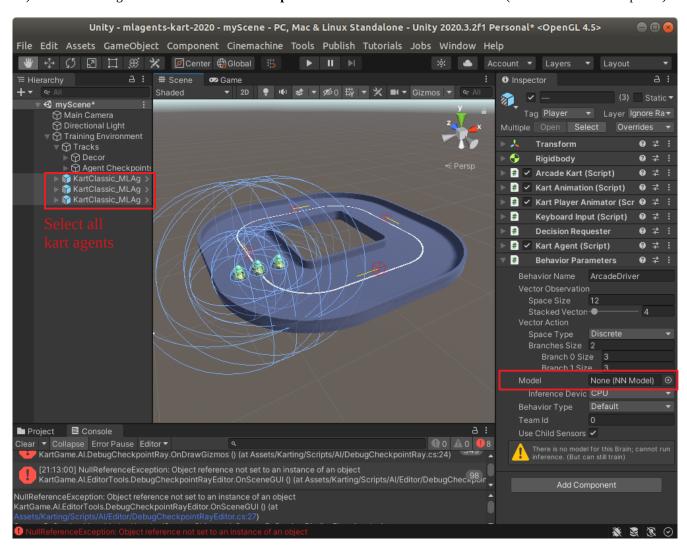
- 26) Make sure **Checkpoint Mask** from **Kart Agent** is set to **TrainingCheckpoints** (shown above in yellow box)
- 27) Place KartClassic\_MLAgent under Training Environment in Heirarchy



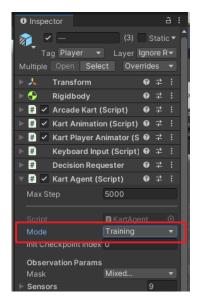
28) Duplicate the **KartClassic MLAgent** a few time to train the brain faster (too much will slow down simulation)



29) Select all kart agents and choose **None** for **Inspector** > **Behavior Parameters** > **Model** (click on circle to see options)



#### 30) In **Inspector** > **Kart Agent**, change **Mode** to **Training** for all kart agents



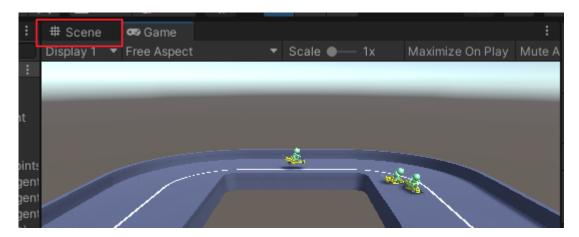
- 31) Go back to terminal and enter the following command to start training
- cd ~/<Project-Path>/
- 2 mlagents-learn Assets/Karting/Prefabs/AI/kart\_mg\_trainer\_config.yaml --train --run-id=custom-track-1

Notes: configuration found in project dir>/Assets/Karting/Prefabs/AI/kart\_mg\_trainer\_config.yaml (you can also open it from Unity through the same path starting from Assets in the Project window)

- use --force to overwrite current run-id
- use --resume to continue previously paused/stopped training identified by run-id
- hidden\_units: number of nodes per layer in NN
- num\_layers: number of layers in NN
- max\_steps: number of training steps (I changed my max steps in to 5.0e5)
- parameter details: <a href="https://github.com/Unity-Technologies/ml-agents/blob/main/docs/Training-Configuration-File.md">https://github.com/Unity-Technologies/ml-agents/blob/main/docs/Training-Configuration-File.md</a>
- 32) If no errors occur, it will prompt user to press Play on Unity Editor



RNING [learn.py:534] The --train option has been deprecated. Train mode is now the default. Use --inference to run in inference mode FO [environment.py:201] Listening on port 5004. Start training by pressing the Play button in the Unity Editor. 33) When **Play** button is pressed it automatically goes to **Game** mode where camera POV is fixed (depending on where you placed your track, the training might not be visible); go to **Scene** mode to move your POV



34) Various statistics and configuration details will be displayed on terminal (the karts will get better as it continues to train)

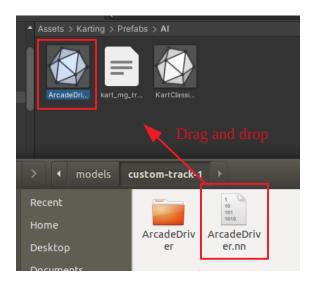
```
2021-04-05 22:50:10 INFO [stats.py:130] Hyperparameters for behavior name custom-track-1 ArcadeDriver:
                                  рро
512
           batch_size:
           beta: 0.005
buffer_size:
                                  10240
           epsilon:
hidden_units:
lambd: 0.95
                                  0.2
128
           learning_rate: 0.0002
learning_rate_schedule: linear
max_steps: 5.0e5
           memory_size:
normalize:
                                  256
                                  False
           num_epoch:
num_layers:
time_horizon:
           sequence_length:
summary_freq:
use_recurrent:
                                              64
                                  1000
                                  False
           vis_encode_type:
reward_signals:
                                              simple
              extrīnsīc:
                 strength:
                                  1.0
0.99
                 gamma:
                mary_path:
                                  custom-track-1_ArcadeDriver
            model_path:
                                  ./models/custom-track-1/ArcadeDriver
            keep checkpoints:
```

```
Time Elapsed: 37.162 s Mean Reward:
Time Elapsed: 37.158 s Mean Reward:
Time Elapsed: 39.588 s Mean Reward:
Time Elapsed: 42.078 s Mean Reward:
Time Elapsed: 44.317 s Mean Reward:
Time Elapsed: 46.732 s Mean Reward:
                                                                                                                                                                                                                                                                                                                                                                                                                                                                   -2.273. Std of Reward:
-2.323. Std of Reward:
-2.336. Std of Reward:
-2.175. Std of Reward:
-1.692. Std of Reward:
-2.033. Std of Reward:
2021-04-05 22:50:21 INFO
2021-04-05 22:50:26 INFO
                                                                                                 [stats.py:111]
[stats.py:111]
[stats.py:111]
                                                                                                                                                      custom-track-1_ArcadeDriver: Step: 5000.
custom-track-1_ArcadeDriver: Step: 6000.
                                                                                                                                                      custom-track-1_ArcadeDriver: Step: 6000.
custom-track-1_ArcadeDriver: Step: 7000.
custom-track-1_ArcadeDriver: Step: 8000.
custom-track-1_ArcadeDriver: Step: 9000.
custom-track-1_ArcadeDriver: Step: 10000.
custom-track-1_ArcadeDriver: Step: 11000.
custom-track-1_ArcadeDriver: Step: 12000.
custom-track-1_ArcadeDriver: Step: 13000.
custom-track-1_ArcadeDriver: Step: 14000.
                                                                                                [stats.py:111]
[stats.py:111]
[stats.py:111]
[stats.py:111]
[stats.py:111]
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            2.033. Training.
1.490. Training.
1.910. Training.
2021-04-05 22:50:28 INFO
2021-04-05 22:50:30 INFO
2021-04-05 22:50:33 INFO
                                                                                                                                                                                                                                                                                                                          Time Elapsed:
Time Elapsed:
                                                                                                                                                                                                                                                                                                                                                                                49.218 s Mean Reward:
52.331 s Mean Reward:
54.694 s Mean Reward:
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    Std of
Std of
2021-04-05 22:50:35
2021-04-05 22:50:39
                                                                                                                                                                                                                                                                                                10000.
11000.
                                                                                                    stats.py:111
                                                                                                  stats.py:111]
stats.py:111]
                                                                                                                                                                                                                                                                                                                            Time
                                                                                                                                                                                                                                                                                                                                             Elapsed:
```

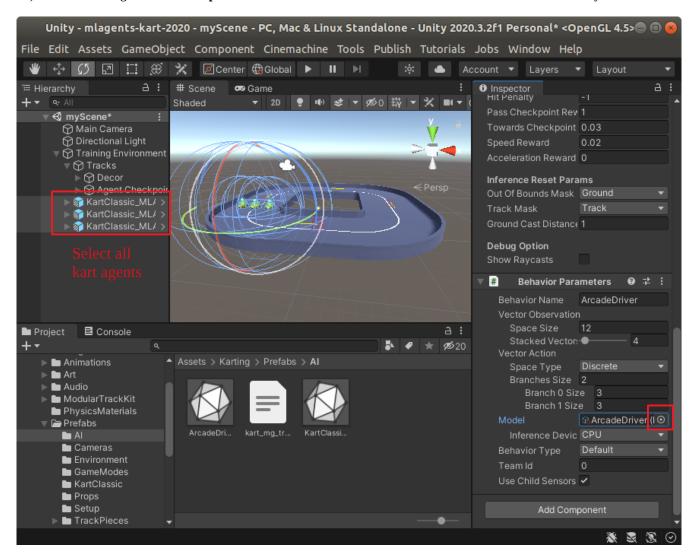
35) Wait for training to finish where the following message appears on terminal

```
2021-04-05 23:08:57 INFO [trainer_controller.py:112] Saved Model
2021-04-05 23:08:57 INFO [model_serialization.py:223] is_continuous_control
2021-04-05 23:08:57 INFO [model_serialization.py:223] is_continuous_control
2021-04-05 23:08:57 INFO [model_serialization.py:223] version_number
2021-04-05 23:08:57 INFO [model_serialization.py:223] memory_size
2021-04-05 23:08:57 INFO [model_serialization.py:223] action_output_shape
2021-04-05 23:08:57 INFO [model_serialization.py:223] action
Converting ./models/custom-track-1/ArcadeDriver/frozen_graph_def.pb to ./models/custom-track-1/ArcadeDriver.nn
GLOBALS: 'is_continuous_control', 'version_number', 'memory_size', 'action_output_shape'
IN: 'vector_observation': [-1, 1, 1, 48] => 'policy/main_graph_0/hidden_0/BiasAdd'
IN: 'action_masks': [-1, 1, 1, 6] => 'policy_1/strided_slice'
IN: 'action_masks': [-1, 1, 1, 6] => 'policy_1/strided_slice_1'
OUT: 'action'
DONE: wrote ./models/custom-track-1/ArcadeDriver.nn file
2021-04-05 23:08:57 INFO [model_serialization.py:76]
Exported ./models/custom-track-1/ArcadeDriver.nn file
(mlkart20) chee@cheeUbuntu:~/Documents/mlagents-kart-20205
```

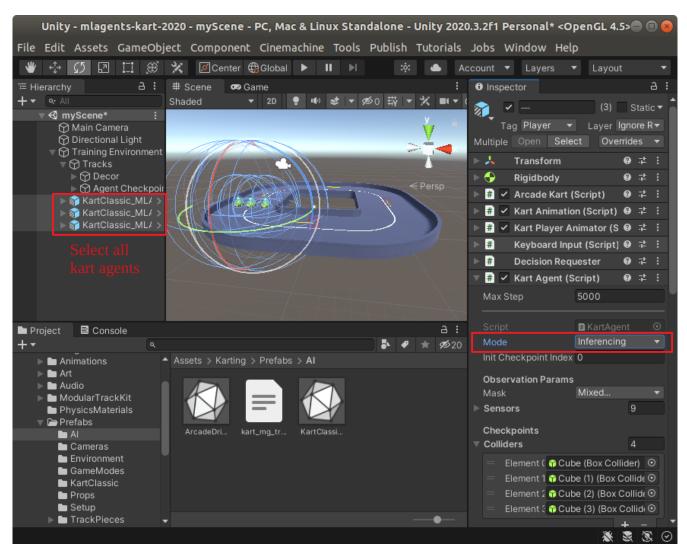
36) Locate trained model file (path found in the terminal in yellow box above); then, drag and drop into **Project** > **Assets** > **Karting** > **Prefabs** > **AI** 



37) Select all kart agents and on **Inspector** > **Behavior Paramters** > **Models** click on circle and select newly trained model



38) While the kart agents are still selected, change Inspector > Kart Agent > Mode to Inferencing



- 39) Press **Play** button to watch the inferencing
- 40) This training was done with PPO. Next, SAC is conducted with the same number of max\_steps.

  - 40.2 Rename the file to **kart\_mg\_trainer\_config\_sac.yaml** and modify its contents



- 40.3 Remove lines 4, 6, 8, 14 (beta, epsilon, lambd, num epoch) since these are PPO-specific configurations
- 40.4 Change **learning\_rate\_schedule** to **constant**

40.5 Add the following SAC-specific configurations to default which can be found <a href="https://github.com/Unity-Technologies/ml-agents/blob/main/docs/Training-Configuration-File.md">https://github.com/Unity-Technologies/ml-agents/blob/main/docs/Training-Configuration-File.md</a>: (Note: do not use TAB, use spaces instead)

```
#SOC-Specific Configurations
buffer_init_steps: 1000
init_entcoef: 0.75
save_replay_buffer: false
tau: 0.005
steps_per_update: 3 # Typically, equal to # agents
reward_signal_num_update: steps_per_update
```

40.6 Resulting configuration file (kart\_mg\_trainer\_config\_sac.yaml) is:

```
kart_mg_trainer_config_sac.yaml
 Open ▼
          æ
 1 default:
 2
      trainer: sac # change trainer
      batch_size: 1024
 3
 4
      buffer_size: 10240
 5
      hidden_units: 128
 б
      learning_rate: 3.0e-4
 7
      learning_rate_schedule: constant # SAC recommended
 8
      max_steps: 5.0e5
 9
      memory_size: 256
10
      normalize: false
11
      num_layers: 2
      time_horizon: 64
12
      sequence_length: 64
13
      summary_freq: 1000
14
15
      use_recurrent: false
      vis_encode_type: simple
16
17
      reward_signals:
18
           extrinsic:
19
               strength: 1.0
20
               gamma: 0.99
21
22
      #SOC-Specific Configurations
23
      buffer_init_steps: 1000
24
      init_entcoef: 0.75
25
      save_replay_buffer: false
26
      steps_per_update: 3 # Typically, equal to # agents
27
28
      reward_signal_num_update: steps_per_update
29
30 ArcadeDriver:
      batch_size: 512
31
      learning_rate: 2.0e-4
32
33
34 4x4Driver:
35
      beta: 5.0e-4
36
      batch_size: 512
37
38 MuscleDriver:
39
      batch_size: 512
40
41 RoadsterDriver:
42
      batch_size: 512
43
      learning_rate: 2.0e-4
```

40.7 Repeat steps 28-38 for training and inferencing with SAC configurations, except for step 30 use the commands

```
1 cd ~/<Project-Path>/
2 mlagents-learn Assets/Karting/Prefabs/AI/kart_mg_trainer_config_sac.yaml --train --run-id=custom-track-sac
```

#### 41) Comparing results on TensorBoard

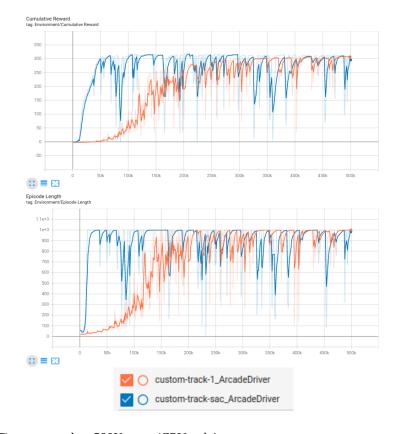
The tensorboard event files are saved into the summaries folder in the project directory. To visualize the results, enter

```
1 cd /<Project-Path>/
2 tensorboard --logdir summaries --port 6006
```

and go to the link <a href="http://localhost:6006/">http://localhost:6006/</a> in a web browser

```
(mlkart20) chee@cheeUbuntu:~/Documents/mlagents-kart-2020$ tensorboard --logdir results --port 6006
2021-04-06 14:36:45.718479: W tensorflow/stream_executor/platform/default/dso_loader.cc:60] Could not load dynamic
library 'libcudart.so.11.0'; dlerror: libcudart.so.11.0: cannot open shared object file: No such file or directory;
LD_LIBRARY_PATH: /opt/ros/melodic/lib
2021-04-06 14:36:45.718502: I tensorflow/stream_executor/cuda/cudart_stub.cc:29] Ignore above cudart dlerror if you
do not have a GPU set up on your machine.
Serving TensorBoard on localhost; to expose to the network, use a proxy or pass --bind_all
TensorBoard 2.4.1 at http://localhost:6006/ (Press CTRL+C to quit)
```

It will display a few graphs, where one contains the **Cumulative Rewards** 



Cumulative reward is the mean cumulative episode reward over all agents

We can see that both PPO and SAC goes up to a mean reward around 300. However, SAC (blue) reached that value much quicker at 50K steps while PPO took around 300K steps. From these graphs, the max\_steps needs to be increased to have a more stable result.

More information about the graphs is at <a href="https://github.com/Unity-Technologies/ml-agents/blob/main/docs/Using-Tensorboard.md">https://github.com/Unity-Technologies/ml-agents/blob/main/docs/Using-Tensorboard.md</a>

Time to complete 500K steps (CPU only):

- PPO: Time elapsed = 1148.251 s = 19 min
- SAC: Time elapsed = 5453.848 s = 90 min

(END)