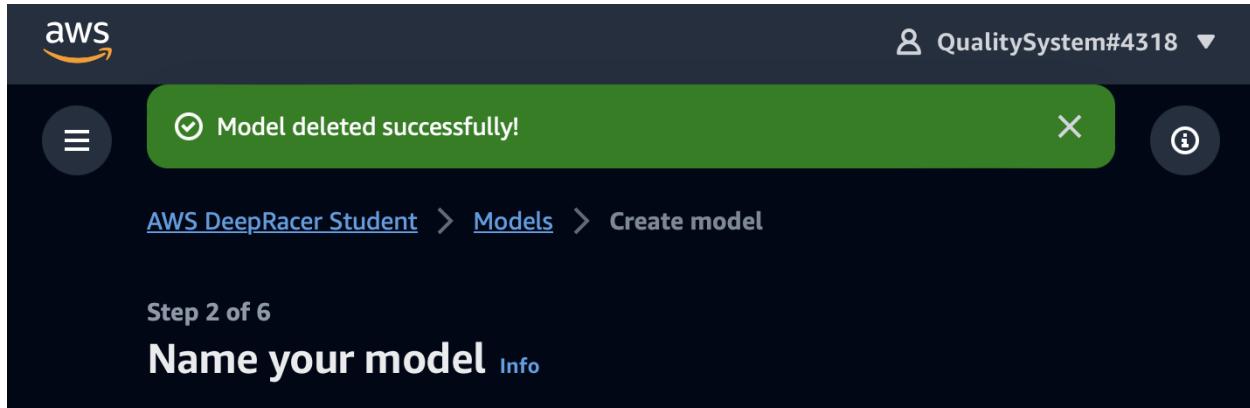


Lab 12

By Vannarith Om

Setting up the name of the first model.



To get started, give your model a unique name. This will help you find your model later when you want to clone and improve it. Be specific and descriptive. You cannot change the name of your model after training has begun.

Model name

LocoSlug

The model name can have up to 64 characters. Valid characters are a-z, A-Z, 0-9, and - (hyphen). No spaces or underscores.

[Cancel](#)

[Previous](#)

[Next](#)

Selecting re:Invent 2018 track the track to train the model



Po-Chun Speedway

Direction: Clockwise,
Counterclockwise



Smile Speedway

Direction: Clockwise,
Counterclockwise



re:Invent 2018

Direction: Counterclockwise



A to Z Speedway

Direction: Clockwise,
Counterclockwise

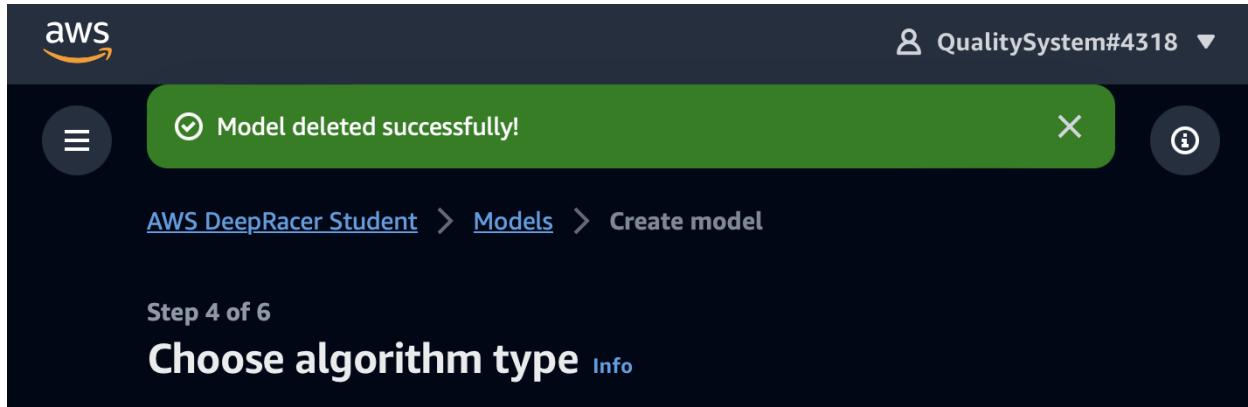


[Cancel](#)

[Previous](#)

[Next](#)

Selecting the the Proximal Policy Optimization (PPO) as the algorithm type.



A model training algorithm is a procedure that uses data to create a model. These algorithms maximize total reward differently. Proximal Policy Optimization (PPO) explores the environment less compared to Soft Actor Critic (SAC).

Choose an algorithm for training your model. We encourage you to experiment with both the algorithms.

Proximal Policy Optimization (PPO)

Needs more data to produce consistent results

Soft Actor Critic (SAC)

Uses less data but produces less consistent results

Cancel

Previous

Next

Initial Reward system that balances following the middle line and maintaining the car inside the border lines. Confirming that the reward function can be validated.

```
1 def reward_function(params):
2     # Constants
3     MAX_SPEED = 5.0
4     MIN_SPEED = 2.0
5     MAX_STEERING_ANGLE = 30.0
6
7     # Read input parameters
8     distance_from_center =
9         params['distance_from_center']
10    all_wheels_on_track =
11        params['all_wheels_on_track']
12    speed = params['speed']
13    steering_angle = abs(params['steering_angle'])
14
15    # Calculate reward
16    reward = 0.0
17
18    # Stay close to the centerline
19    if distance_from_center < 0.1:
20        reward += 1.0
21    else:
22        reward += 0.5
23
24    # Stay within the track borders
25    if all_wheels_on_track:
26        reward += 1.0
27    else:
28        reward -= 1.0
29
30    # Speed reward
31    reward += min(speed/MAX_SPEED, 1.0)
32
33    # Penalize steering too much
34    if steering_angle > MAX_STEERING_ANGLE:
35        reward -= 0.5
36
37
38    return float(reward)
```

Python

Ln 1, Col 1

✖ Errors: 0

⚠ Warnings: 0



```
17▼    if distance_from_center < 0.1:
18        reward += 1.0
19▼    else:
20        reward += 0.5
21
22    # Stay within the track borders
23▼    if all_wheels_on_track:
24        reward += 1.0
25▼    else:
26        reward -= 1.0
27
28    # Speed reward
29    reward += min(speed/MAX_SPEED, 1.0)
30
31    # Penalize steering too much
32▼    if steering_angle > MAX_STEERING_ANGLE:
33        reward -= 0.5
34
35    return float(reward)
```

Python

Ln 1, Col 1

✖ Errors: 0

⚠ Warnings: 0



Reset

Validate

Once you are done with changes, click Next.

✓ Reward function successfully validated!

Cancel

Previous

Next

Setting the initial training to 60 minutes (one hour)

ⓘ The training is being initialized. X

This can take about 6 minutes. You will shortly see how the AWS DeepRacer car (agent) is being trained on the racetrack (environment).

[AWS DeepRacer Student](#) > [Models](#) > [LocoSlug](#)

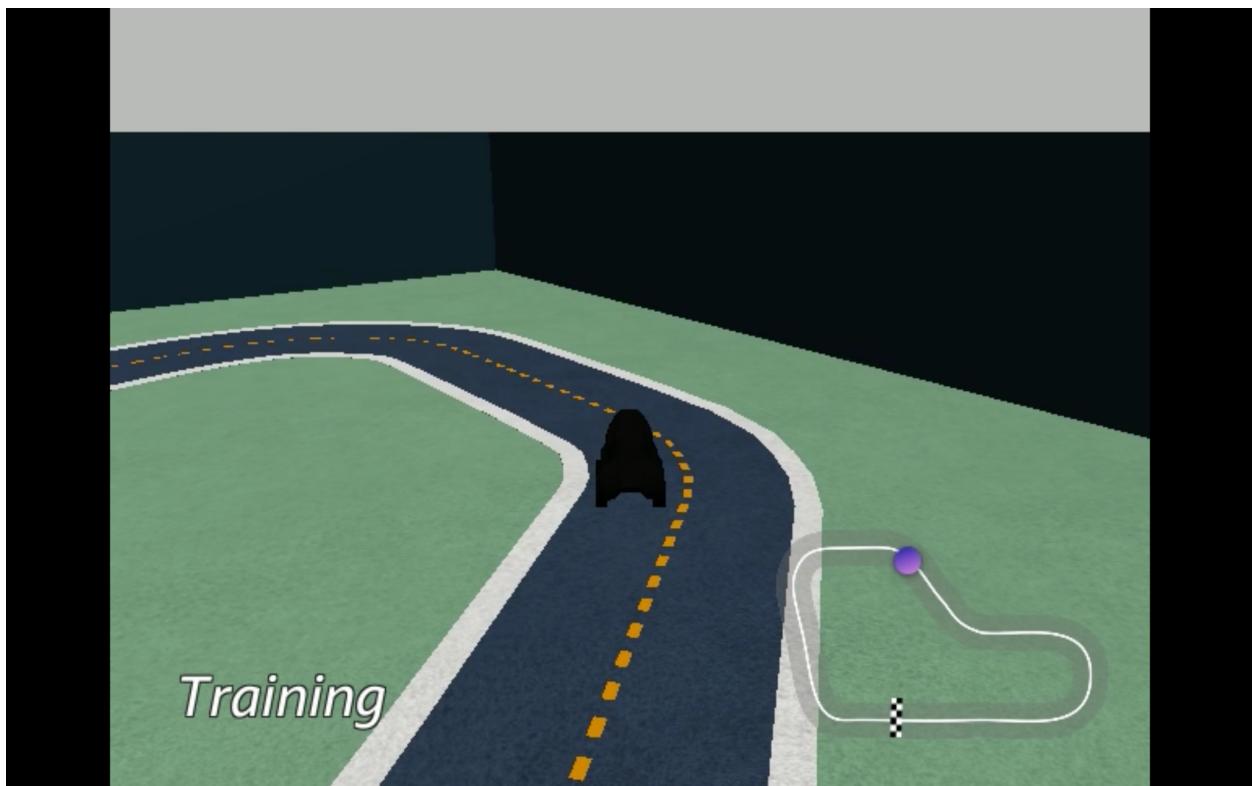
LocoSlug Info

Training configuration <small>Info</small>				
Track	Algorithm	Model description	Training time remaining (hh:mm)	Status
re:Invent 2018 - Counterclock wise	PPO	Stay focus	01:00	⌚ Created

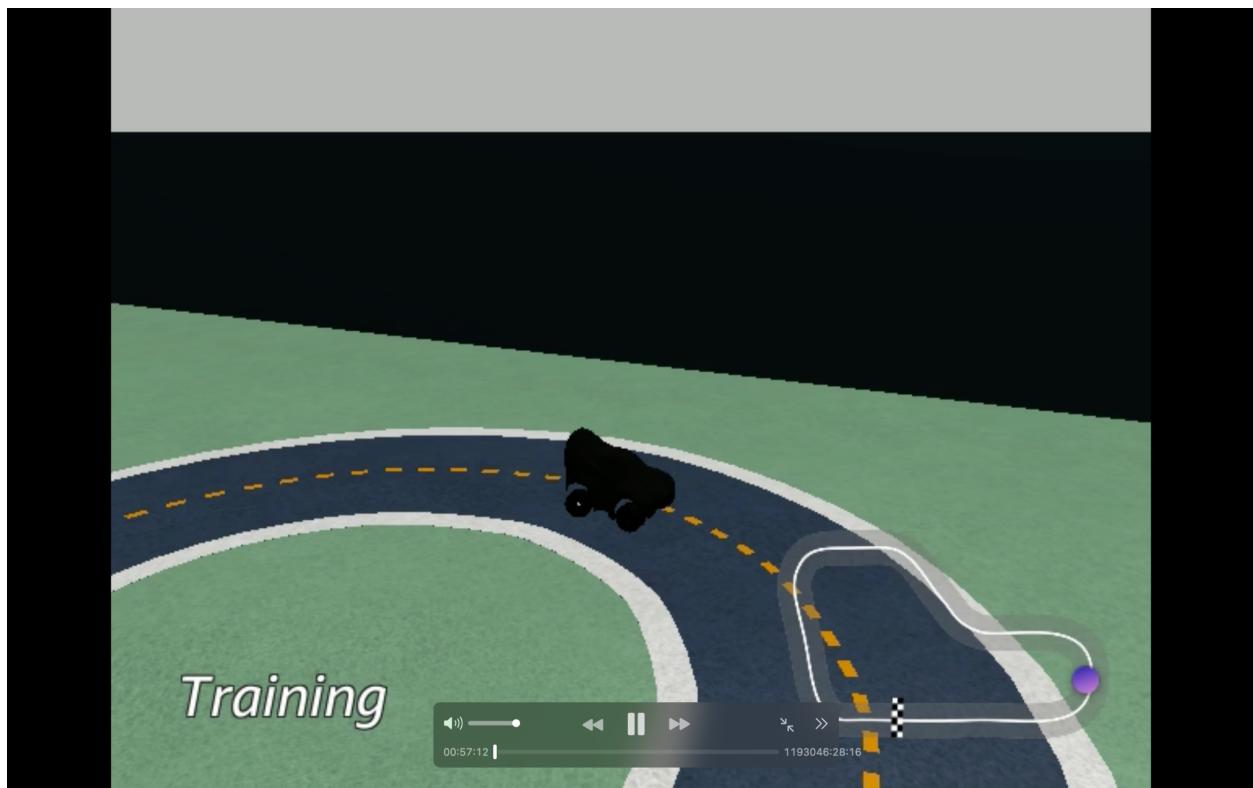
Simulation video stream

Simulation video stream is not available. Video is only available during training.

Car during first minutes of training process.



Car doing getting better at curves as time of training advances.



LocoSlug being evaluated for the preseason practice session

The screenshot shows the AWS DeepRacer 2024 Preseason practice session interface. At the top, there's a navigation bar with icons for back, forward, search, and refresh. The main title is "2024 Preseason practice session". Below the title, there's a "Race overview" section with details like race type (Time trial), competition track (Ostro Raceway), and rules (Total time, individual lap, 3 consecutive laps, permitted resets). A "Race date" section shows the start date as September 30 at 5:00 PM and end date as February 29 at 5:59 PM. A "Time zone" section indicates UTC-0800 (Pacific Standard Time) America/Los_Angeles.

In the center, there's a "LocoSlug evaluation video" window showing a 3D simulation of the race track. The video frame displays the text "QualitySystem#4 2/3" and "00:78 m/s 2024 Preseason practice session Time trial race". To the right of the video, three performance metrics are listed: "Gap to 1st +00:46.988", "Gap to 1st +00:53.800", and "Gap to 1st +01:02.356".

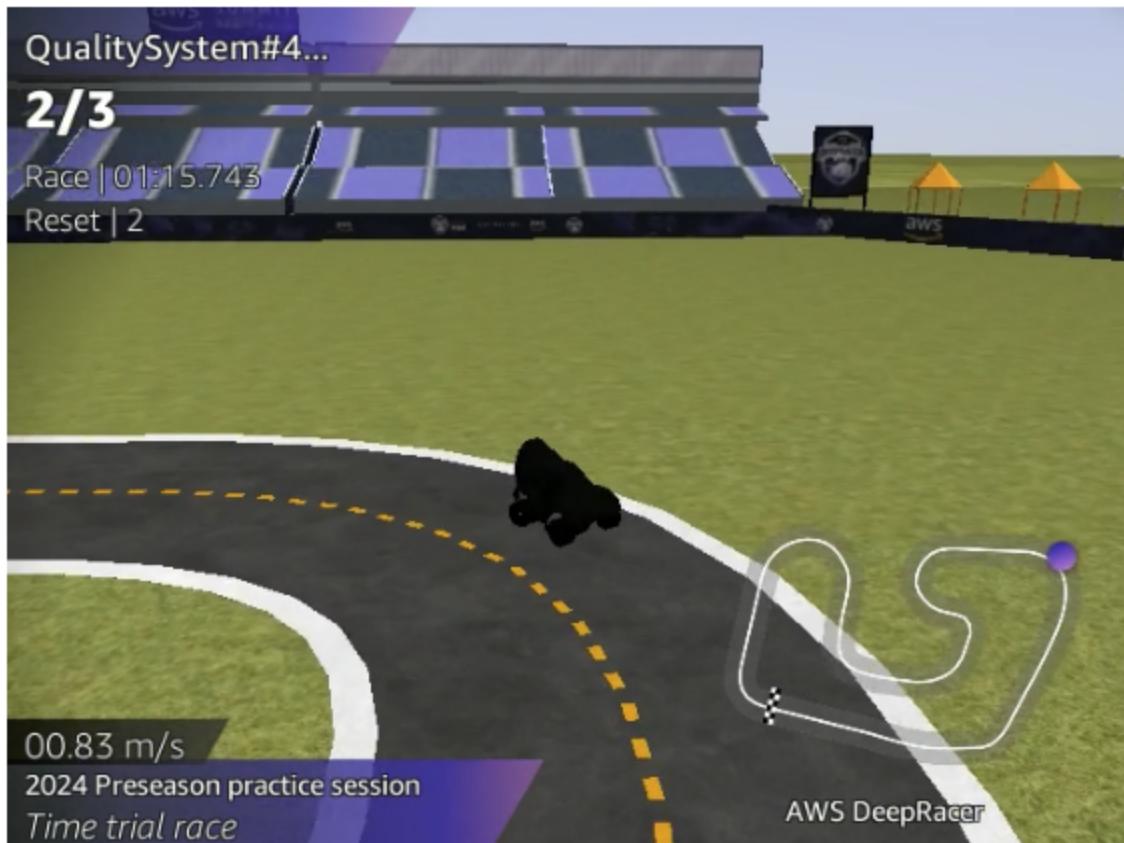
On the left, there's a profile card for "QualitySystem#4318" with a picture of a person wearing a racing helmet. It shows the fastest time as "02:46.730", the fastest model submitted as "LocoSlug", and remaining submissions as "249".

At the bottom, there's a "Race leaderboard" and a "Your submissions (2)" section. The "Your submissions" table has columns for Model name, Time, Status, Date submitted to race, and Video. Two entries are listed: "LocoSlug" with status "Evaluating" and "Submitted" dates of 11/15/2023, 12:19 PM and 11/15/2023, 11:46 AM respectively, and "Video" status "Not available".

At the very bottom, there are links for Privacy, Site Terms, and Cookie Preferences, along with a copyright notice: "© 2022 Amazon Web Services, Inc. or its affiliates. All rights reserved."

LocoSlug evaluation video

X



Copy of the original model with implementation of the Markov Decision Process in the rewarding funcion.

AWS DeepRacer Student > Models > Clone model

Step 1 Name your model
Step 2 **Customize reward function**
Step 3 Choose duration

Customize reward function Info

What is a reward function?
The reward function is Python code that describes immediate feedback in the form of a reward or penalty to move from a given position on the track to a new position.

What is the purpose of a reward function?
The reward function encourages the vehicle to make moves along the track quickly to reach its destination.

Choose and modify one of the sample reward functions.

Follow the centerline
Sections the track into three reward zones. The farther the car strays from the centerline, the less it's rewarded.


Stay within borders
Defines the track border and gives high rewards if all 4 wheels stay on the track, which leaves it to the model to find the optimal path.


Prevent zig-zag
Is the centerline reward function plus a steering angle range limit. Keeps the car driving smoothly down the middle.


```
1 def reward_function(params):
2     # Constants
3     MAX_SPEED = 5.0
4     MIN_SPEED = 2.0
5     MAX_STEERING_ANGLE = 30.0
6     REWARD_SCALE = 100.0 # Adjust as needed
7
8     # Read input parameters
9     distance_from_center = params['distance_from_center']
10    all_wheels_on_track = params['all_wheels_on_track']
11    speed = params['speed']
12    steering_angle = abs(params['steering_angle'])
13
14    # Calculate reward components
15    centerline_reward = 1.0 if distance_from_center < 0.1 else 0.5
16    borders_reward = 1.0 if all_wheels_on_track else -1.0
17    speed_reward = min(speed / MAX_SPEED, 1.0)
18    steering_penalty = 0.5 if steering_angle > MAX_STEERING_ANGLE else 0.0
19
20    # Combine rewards using the Bellman Equation
21    reward = (
22        REWARD_SCALE * (centerline_reward + borders_reward + speed_reward
23        - steering_penalty)
24    )
25
26    return float(reward)
```

Python Ln 15, Col 25 Errors: 0 Warnings: 0