



デベロッパーガイド

AWS DeepRacer



AWS DeepRacer: デベロッパーガイド

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon の商標およびトレードドレスは Amazon 以外の製品およびサービスに使用することはできません。また、お客様に誤解を与える可能性がある形式で、または Amazon の信用を損なう形式で使用することもできません。Amazon が所有していないその他のすべての商標は Amazon との提携、関連、支援関係の有無にかかわらず、それら該当する所有者の資産です。

Table of Contents

AWS DeepRacer とは何ですか?	1
AWS DeepRacer コンソール	1
AWS DeepRacer 車両	2
AWS DeepRacer リーグ	2
強化学習を探索する	3
概念と用語	4
レースイベントの用語	9
仕組み	11
強化学習	11
アクションスペースと報酬関数	13
トレーニングアルゴリズム	16
AWS DeepRacer のワークフロー	17
Simulated-to-Real パフォーマンスギャップ	19
はじめに	20
最初のモデルのトレーニング	20
AWS DeepRacer コンソールを使用して強化学習モデルをトレーニングします。	20
モデル名と環境を指定します。	20
レースタイプとトレーニングアルゴリズムを選択します。	21
アクションスペースの定義	23
バーチャルカーを選択	27
報酬機能をカスタマイズします。	28
シミュレーションでのモデルの評価	31
モデルのトレーニングと評価	35
レースの種類とセンサーの有効化	36
センサーを選択する	37
トレーニング用にエージェントを設定する	39
タイムトライアル向けのレーシングのカスタマイズ	41
オブジェクト回避レース向けのトレーニングのカスタマイズ	42
head-to-bot レース向けのトレーニングのカスタマイズ	44
AWS DeepRacer コンソールを使用したトレーニングと評価	45
報酬関数を作成する	46
アクションスペースを探す	48
ハイパーパラメータの調整	50
トレーニングジョブの進行状況を調べる	56

トレーニング済みモデルの複製	58
シミュレーションでのモデルの評価	59
実環境のトレーニングを最適化する	59
報酬関数リファレンス	62
報酬関数の入力パラメータ	63
報酬関数の例	78
.....	83
AWS DeepRacer モデルを Amazon S3 にコピーする	83
AWS DeepRacer モデルをコンソールにインポートする	86
トラブルシューティング	88
車両を運転する	91
車両について理解する	91
車両を検査する	92
バッテリーを充電して取り付ける	94
コンピューティングモジュールをテストする	96
デバイスをオフにする	97
LED インジケータ	97
デバイスのスペアパーツ	100
車両のセットアップ	111
Wi-Fi を設定する準備を整える	111
Wi-Fi を設定し、ソフトウェアを更新する	112
デバイスコンソールを起動する	113
車両をキャリブレーションする	115
モデルをアップロードする	123
車両を運転する	124
AWS DeepRacer 車両を手動で運転する	125
AWS DeepRacer 車両を自律運転する	126
車両設定を確認、管理する	127
車両ログを表示する	133
AWS DeepRacer デバイスを更新して復元する	135
デバイスのソフトウェアバージョンをチェックする	135
Ubuntu 20.04 のインストールメディアを作成する	136
前提条件	136
準備	136
ブート可能な USB ドライブを準備する	137
Ubuntu 20.04 にデバイスを更新します	148

物理的なトラックを構築する	150
マテリアルとツール	150
必要なマテリアル	150
必要なツール	151
トラックの配置	151
寸法の要件	152
パフォーマンスに関する考慮事項	153
トラックを構築するステップ	154
トラック設計テンプレート	158
A から Z までのスピードウェイ (ベーシック) トラックのテンプレート	159
AWS DeepRacer Smile Speedway (中級) トラックのテンプレート	160
RL スピードウェイ (上級) トラックのテンプレート	161
シングルターントラックのテンプレート	162
S カーブトラックのテンプレート	163
ループトラックのテンプレート	164
レースに参加する	167
レースイベントの種類	167
オンライン AWS スポンサーまたはコミュニティスポンサーのレースへの参加	167
仮想サーキットレースに参加する	168
コミュニティレースに参加する	169
レース参加者として AWS DeepRacer コミュニティレースに参加します。	170
LIVE レースに参加する	178
レースの開始	182
.....	182
レースを作成する (クイックスタート)	182
レースをカスタマイズする	187
LIVE レースで走行する	193
LIVE レースを放送する	199
主催者ロール	199
ブロードキャスターのシーン	199
AWS DeepRacer シーンテンプレート	200
レースの管理	205
イベントの主催	209
AWS DeepRacer イベントとは何ですか?	209
イベントの仕組みと期待する内容	209
始める前に考慮すべきこと	210

AWS DeepRacer レースの種類	212
ベストプラクティス	213
イベント入門	213
AWS DeepRacer イベントの例	214
その他のリソース	218
マルチユーザー モード	219
管理者セットアップ	219
マルチユーザーの利害関係者	220
ステップ 1: AWS DeepRacer マルチユーザーモードの前提条件	220
ステップ 2: マルチユーザーアカウントモードを有効にします。	222
ステップ 3: 参加者をスポンサーに招待する	223
ステップ 4: 使用量クォータを設定する	224
ステップ 5: 使用状況のモニタリング	224
次の手順	226
参加者の設定	227
前提条件	227
ステップ 1. スポンサーアカウントの認証情報を使用して AWS コンソールにログインする	228
ステップ 2. AWS Player アカウントを作成またはログインする	228
ステップ 3. プロフィールのカスタマイズ。	229
ステップ 4. モデルをトレーニングします。	229
ステップ 5. スポンサーの利用状況を表示します	230
ステップ 6. (オプション) スポンサー時間の追加をリクエストします	230
教育ツール	231
AWS DeepRacer Student を教室に組み込む	231
学生コミュニティレースの作成	231
学生レースの作成	232
学生レースのカスタマイズ	233
学生レースの管理	236
セキュリティ	239
データ保護	239
AWS DeepRacer に依存するサービス	240
必須 IAM ロール	242
AWS Identity and Access Management	243
オーディエンス	243
アイデンティティを使用した認証	243

ポリシーを使用したアクセスの管理	245
How AWS DeepRacer と IAM の連携	247
アイデンティティベースのポリシーの例	252
AWS マネージドポリシー	255
サービス間での不分別な代理処理の防止	259
トラブルシューティング	261
Tagging	265
新規リソースへタグの追加、表示、編集を行う。	266
既存のリソースにタグの追加、表示、編集を行う	268
一般的な問題のトラブルシューティング	270
AWS DeepRacer LIVE の一般的な問題を解決する方法	270
LIVE レースのページでレースのビデオが見れません。	270
レースキューのレーサーの名前は赤です。	271
LIVE レースを起動しているのに、レーサーを立ち上げられません。	272
Chrome または Firefox のブラウザ使っていますが、まだ LIVE レースを見ることができません。	273
コンピュータと車両の間で、デバイスコンソールと USB を接続できないのはなぜですか?	274
AWS DeepRacer コンピューティングモジュールの電源を、バッテリーからコンセントに切り替える方法	278
USB フラッシュドライブを使用して AWS DeepRacer を Wi-Fi ネットワークに接続する方法	279
車両のドライブモジュールのバッテリーを充電する方法	284
車両のコンピューティングモジュールのバッテリーを充電する方法	288
バッテリーは充電されていますが、車両が動きません	288
車両バッテリーのロックアウトのトラブルシューティング	291
車両のバッテリーのロックアウトを防ぐ方法	292
AWS DeepRacer 車両のバッテリーをロック解除する方法	292
LiDAR センサーの取り付け時に Dell バッテリーコネクタケーブルを巻きつける方法	295
車両の接続を維持する方法	300
車両の Wi-Fi の LED インジケータが青色に点滅し、その後 2 秒間赤色に変わって、最後にオフになった場合の Wi-Fi 接続のトラブルシューティング方法	300
車両の Wi-Fi または電源 LED インジケータが青く点滅しているのはどういう意味ですか?	301
ホスト名を使用して車両のデバイスコンソールに接続するにはどうすればよいですか?	302
IP アドレスを使用して車両のデバイスコンソールに接続する方法	302
デバイスの Mac アドレスを取得する方法	302

デバイスコントローラのデフォルトパスワードを回復する方法	303
デバイスを手動で更新する方法	305
デバイスの一般的な操作問題の診断および解決方法	306
デバイスコンソールのビデオプレーヤーに車両のカメラからのビデオストリームが表示されないのはなぜですか?	307
なぜ、私の AWS DeepRacer が 動かないのでしょうか?	307
デバイスの最新アップデートが表示されないのはなぜですか? 最新アップデートを入手する方法を教えてください。	308
AWS DeepRacer 車両が Wi-Fi ネットワークに接続されないのはなぜですか。	308
AWS DeepRacer のデバイスコンソールページの読み込みに時間がかかるのはなぜですか。	308
AWS DeepRacer 車両にデプロイすると、モデルが適切に動作しないのはなぜですか?	309
ドキュメント履歴	310
AWS 用語集	313
.....	cccxiv

AWS DeepRacer とは何ですか？

AWS DeepRacer は、「[強化学習](#)」によって駆動される完全自律型の 1/18 スケールのレースカーです。これは次のコンポーネントで構成されます。

- AWS DeepRacer コンソール: シミュレートされた自動運転環境で[強化学習モデルをトレーニングし](#)、評価する [AWS Machine Learning](#) サービスです。
- AWS DeepRacer 車両: 自動運転用にトレーニングされた [AWS DeepRacer モデルで推論を実行できる](#) 1/18 スケールの RC 車両です。
- AWS DeepRacer: 世界初のグローバルな自律型レースリーグです。賞品、栄光、ワールドチャンピオンカップへの出場権を得るためのレースです。詳細については、「[用語と条件](#)」を参照してください。

トピック

- [AWS DeepRacer コンソール](#)
- [AWS DeepRacer 車両](#)
- [AWS DeepRacer リーグ](#)
- [AWS DeepRacer を使用して強化学習を試す](#)
- [AWS DeepRacer の概念と用語](#)

AWS DeepRacer コンソール

AWS DeepRacer コンソールは、AWS DeepRacer サービスのインタラクティブなグラフィカルユーザーインターフェイスです。コンソールを使用して、強化学習モデルをトレーニングしたり、AWS DeepRacer シミュレーターでモデルパフォーマンスを評価したりできます。コンソールでは、トレーニングされた強化学習モデルをダウンロードして、物理環境で自動走行する AWS DeepRacer 車両にデプロイできます。

要約すると、AWS DeepRacer コンソールは次の機能をサポートしています。

- 指定された報酬関数、最適化アルゴリズム、環境、ハイパーパラメータを使用して強化学習モデルをトレーニングするための、トレーニングジョブを作成します。
- シミュレートされたトラックを選択して、SageMaker AI を使用してモデルをトレーニングおよび評価します。

- モデルのパフォーマンスを最適化するためにハイパーパラメータを調整して、トレーニングを向上させるためにトレーニング済みモデルを複製します。
- デプロイのため、トレーニング済みモデルを AWS DeepRacer 車両にダウンロードして、その車両の物理環境での走行を可能にします。
- 自分のモデルを仮想レースに参加させて、そのパフォーマンスを仮想リーダーボードの他のモデルと比較してランク付けします。

AWS DeepRacer サービスコンソールを使用すると、モデルのトレーニング、評価、およびモデルの保存に関する使用量に基づいて料金が請求されます。

AWS DeepRacer の使用を開始するため、初回の AWS DeepRacer ユーザー向けに[無料利用枠](#)が用意されています。これにより、最初のモデルのトレーニングとチューニングを行って、AWS DeepRacer リーグに参加するのに十分な時間があります。AWS DeepRacer リーグ仮想イベントに参加するモデルを送信するためのコストはかかりません。

料金の詳細については、「[AWS DeepRacer サービスの詳細ページ](#)」を参照してください。

AWS DeepRacer 車両

AWS DeepRacer 車両は、Wi-Fi 対応で強化学習モデルを使用して物理的なトラックを走行できる物理的な車両です。

- 自動走行するように、手動で車両を制御したり、車両モデルをデプロイしたりできます。
- 自動モードは、車両のコンピューティングモジュールで推論を実行します。推論は、前面に取り付けられているカメラからキャプチャされた画像を使用します。
- Wi-Fi 接続により、車両はソフトウェアをダウンロードできます。この接続により、ユーザーは、コンピュータまたはモバイルを使用してデバイスコンソールにアクセスして、車両を操作することもできます。

AWS DeepRacer リーグ

AWS DeepRacer リーグは、AWS DeepRacer の重要なコンポーネントです。AWS DeepRacer リーグは、コミュニティと競争を促進することを意図しています。

AWS DeepRacer リーグにより、物理的または仮想のレースイベントで、他の AWS DeepRacer デベロッパーと比較して、開発に取り組むことができます。賞品や実績を獲得する機会だけでなく、

強化学習モデルを測定する方法も用意しています。他の参加者と競い合い、お互いから学び、互いに刺激し合うことができます。AWS DeepRacer League で受け取ったパフォーマンスの実績は、[プロフィール] ページからソーシャルメディアで共有できます。詳細については、「[用語と条件](#)」を参照してください。

[League のレースに参加しましょう。または、モデルのトレーニング方法を学びましょう。](#)

AWS DeepRacer を使用して強化学習を試す

強化学習、特に深層強化学習は、幅広い自律的な意思決定問題を解決するのに効果的であることが証明されています。いくつか例を挙げると、金融取引、データセンターの冷却、フリート実務体制管理、自動運転レースでの用途があります。

強化学習は、実社会の問題を解決する可能性を秘めています。ただし、必要とする技術面の範囲が広く深いため、学習曲線は急になります。実世界の実験では、自走型レーシングカーなどの物理的なエージェントを構築することが必要になります。また、自動車道や公道などの物理的な環境を確保することも必要です。環境には費用がかかり、危険で、時間もかかる可能性があります。これらの要件は単に強化学習を理解することを超えています。

学習曲線を緩やかにするために、AWS DeepRacer は 3 つの方法でプロセスを簡略化します。

- 強化学習モデルをトレーニングし、評価するためのステップごとのガイダンスを提供します。このガイダンスには、定義済みの環境、状態、アクション、カスタマイズ可能な報酬関数が含まれています。
- 仮想[エージェント](#)と仮想環境の間のやり取りをエミュレートするシミュレーターを提供する。
- AWS DeepRacer 車両を物理エージェントとして提供します。車両を使用して、物理的環境でトレーニングされたモデルを評価します。これは、実際のユースケースに非常に類似しています。

経験豊富な機械学習実務者にとって、AWS DeepRacer は、仮想環境と物理環境の両方で自律型レースのための強化学習モデルを構築する、歓迎すべき機会となるでしょう。要約すると、AWS DeepRacer を使用して、次のステップで自走型レースのための強化学習モデルを作成します。

1. 自走型レースのため、カスタム強化学習モデルをトレーニングします。これを行うには、SageMaker AI と統合された AWS DeepRacer コンソールを使用します。SageMaker
2. AWS DeepRacer シミュレーターを使用して、モデルを評価し、仮想環境で自走型レースをテストします。

3. トレーニングされたモデルを AWS DeepRacer モデル車両にデプロイして、自走型レースを物理環境でテストします。

AWS DeepRacer の概念と用語

AWS DeepRacer は次の概念に基づいて構築されており、次の用語を使用しています。

AWS DeepRacer サービス

AWS DeepRacer は AWS Machine Learning で自律型レーシングに特化した強化学習を探求するための、AWS 機械学習サービスです。AWS DeepRacer サービスは以下の特徴をサポートしています。

1. クラウド上で強化学習モデルをトレーニングする。
2. AWS DeepRacer コンソールでトレーニング済みモデルを評価する。
3. トレーニング済みモデルを仮想化レースに送信し、資格がある場合は、そのパフォーマンスをイベントのリーダーボードに投稿してもらう。
4. トレーニング済みモデルのクローンを作成して、パフォーマンスを向上させるためのトレーニングを続けます。
5. AWS DeepRacer 車両にアップロードするためのトレーニング済みモデル Artifact をダウンロードします。
6. 自動運転のために車両を物理的なトラック上に置き、モデルを実際のパフォーマンスで評価します。
7. 不要な料金が発生しないように不要なモデルを削除する。

AWS DeepRacer

「AWS DeepRacer」は、次の 3 つの異なる車両を参照できます。

- 仮想レースカー は、オリジナルの AWS DeepRacer デバイス、Evo デバイス、または AWS DeepRacer リーグ仮想サーキットレースの形をとることでさまざまなデジタル報酬を獲得することができます。仮想レースカーの色を変更してカスタマイズすることもできます。
- オリジナルの AWS DeepRacer デバイス は、物理的な 1/18 スケールのモデルカーです。車載カメラとオンボードコンピューティングモジュールを装備しています。コンピューティングモジュールはトラックに沿って自律走行するための推論を実行します。コンピューティングモジュールと車両のシャーシは、それぞれコンピューティングバッテリーとドライブバッテリーとして知られる専用のバッテリーから電力を供給されます。

- AWS DeepRacer Evo デバイスは、オプションのセンサーキットを備えたオリジナルデバイスです。キットには追加のカメラと LIDAR (光検出と測距) が含まれており、車両が背後と側面の物体を検出することができます。このキットには、新しいシエルも含まれています。

強化学習

強化学習とは、環境とのやり取りを通じて、特定の目標を達成するための、エージェントによる自律型決断に焦点を当てた機械学習メソッドです。強化学習では、学習は試行錯誤によって達成され、トレーニングはラベル付きの入力を必要としません。トレーニングは報酬型仮説に基づいていて、一連のアクションの後に将来の報酬を最大化することで、すべての目標を達成できるというものです。強化学習では、報酬関数を設計することが重要です。報酬関数を向上させることで、エージェントはより良い意思決定を行うことができます。

自律レーシングでは、エージェントは車両です。環境には走行ルートや交通状況が含まれます。目標は、車両が事故なく短時間で目的地に到着することです。報酬は、目的地への安全かつ迅速な移動を促進するために使用されるスコアです。スコアは危険で無駄な運転に対してペナルティを科します。

トレーニング中に学習を促進するために、学習エージェントは、報酬につながらない可能性のあるアクションを突き止める必要があります。これは、探索と搾取のトレードオフと呼ばれます。エージェントが誤った目的地に導かれる可能性を低くするか取り除くのに役立ちます。

さらに正式な定義については、Wikipedia の「[強化学習](#)」を参照してください。

強化学習モデル

強化学習モデルとは、「エージェントがエージェントが持つ状態」「エージェントが取ることでできるアクション」「アクションを取ることによって得られる報酬」の3つを規定する行動する環境です。エージェントが、そのアクションを決定するための戦略を、ポリシーと呼びます。ポリシーは環境の状態を入力として受け取り、実行するアクションを出力します。強化学習で多くの場合ポリシーは、深層ニューラルネットワークによって表されます。これを強化学習モデルと呼んでいます。トレーニングジョブごとに1つのモデルが生成されます。トレーニングジョブが早期に停止された場合でも、モデルを生成できます。モデルはイミュータブルです。つまり、モデルは作成後に変更したり上書きしたりすることはできません。

AWS DeepRacer シミュレーター

AWS DeepRacer シミュレーターは仮想環境で、トレーニングを視覚化し、AWS DeepRacer モデルを評価します。

AWS DeepRacer 車両

[AWS DeepRacer](#) を参照してください。

AWS DeepRacer カー

このタイプの [AWS DeepRacer 車両](#) は 1/18 スケールのモデルカーです。

リーダーボード

リーダーボードは、AWS DeepRacer レーシングイベントにおける AWS DeepRacer 車両のパフォーマンスのランク付けリストです。レースは、シミュレートされた環境で行われる仮想的なイベント、または現実の環境で行われる物理的なイベントのいずれでもかまいません。パフォーマンス指標はレースの種類によって異なります。所定のレースのトラックと同じ、または類似のトラックでトレーニングしたモデルを評価した AWS DeepRacer のユーザーが送信した最速ラップタイム、合計時間、または平均ラップタイムとなります。

車両が 3 周連続して完走すると、リーダーボードへのランク付けの対象になります。最初の連続した 3 週の平均ラップタイムがリーダーボードに送信されます。

機械学習フレームワーク

機械学習フレームワークとは、機械学習アルゴリズムの構築に使用されるソフトウェアライブラリです。AWS DeepRacer でサポートされたフレームワークには TensorFlow があります。

ポリシーネットワーク

ポリシーネットワークはトレーニングされるニューラルネットワークです。ポリシーネットワークは、入力としてビデオイメージを取り込み、エージェントの次のアクションを予測します。アルゴリズムによっては、エージェントの現在の状態の値も評価される場合があります。

最適化アルゴリズム

最適化アルゴリズムは、モデルをトレーニングするために使用するアルゴリズムです。教師ありトレーニングでは、重みを更新するための特定の戦略を使用して損失関数を最小化することで、アルゴリズムが最適化されます。強化学習では、特定の報酬関数を使用して予想される将来の報酬を最大化することで、アルゴリズムが最適化されます。

ニューラルネットワーク

ニューラルネットワーク (別名、人工ニューラルネットワーク) は、生物学的システムに基づいて情報モデルを構築するために使用される、接続された単位またはノードの集まりです。各ノードは人工ニューロンと呼ばれ、入力 (刺激) を受け取るという点で生物学的ニューロンを模倣し、入力信号が十分に強い場合にアクティブ化 (アクティベーション) され、入力および活性化に基づく出力を生成します。人工ニューラルネットワークはあらゆる関数の汎用近似として機能するため、機械学習で広く使用されています。学習するための機械を教えるのは、任意の入力と出力の最適関数近似を見つけることとなります。深層強化学習では、ニューラルネットワークはポリシーを表し、しばしばポリシーネットワークと呼ばれます。ポリシーネットワークをトレーニン

グすることは、現在のポリシーに基づいてエクスペリエンスを生成し、続いて新しく生成されたエクスペリエンスで、ポリシーネットワークを最適化することを含むステップを繰り返すこととなります。このプロセスは、特定のパフォーマンスメトリクスが必要な基準を満たすまで続きます。

ハイパーパラメータ

ハイパーパラメータは、ニューラルネットワークの学習性能を制御するアルゴリズム依存の変数です。ハイパーパラメータの例としては、各ステップでの学習において、新しい経験をどれだけカウントするかを制御する学習レートがあります。学習レートが大きくなるほど学習速度は速くなりますが、トレーニング済みモデルの品質が低くなる可能性があります。ハイパーパラメータは経験的なものであり、トレーニングごとに体系的な調整が必要です。

AWS DeepRacer トラック

トラックは AWS DeepRacer 車両が走行するパスまたはコースです。トラックは、シミュレーション環境、現実世界、物理的環境の、いずれかに存在できます。シミュレート環境は、仮想トラックで AWS DeepRacer のモデルをトレーニングするために使用します。AWS DeepRacer コンソールでは、仮想トラックを使用できます。AWS DeepRacer の車両を物理的なコースで走らせるためには、実環境を使用します。AWS DeepRacer League は、イベント参加者に物理的なトラックを提供し、競わせるものです。その状況で AWS DeepRacer 車両を走らせる場合は、自分で物理的トラックを作成する必要があります。独自のトラックを構築する方法についての詳細は、[物理的なトラックを構築する](#) を参照してください。

報酬関数

報酬関数とは、実行されたアクションが、以下のいずれのパフォーマンスの結果になったかをエージェントに伝える学習モデル内のアルゴリズムです。

- 強化する必要がある良い結果。
- 中立的な結果です。
- 推奨されない悪い結果。

報酬関数は強化学習の重要な部分です。特定のアクションに対してインセンティブを与えることで、エージェントが学習する挙動を決定します。ユーザーは Python を使用して報酬関数を提供します。この報酬関数は、最適化アルゴリズムによって強化学習モデルをトレーニングするために使用されます。

経験エピソード

経験エピソードとは、エージェントが特定のスタート地点からトラックを完走するまで、またはトラックから外れるまで、環境からトレーニングデータとして経験を収集する期間です。エピソード

ソードごとに長さが異なる場合があります。これはエピソード、または、経験生成エピソードとも呼ばれます。

経験イテレーション

経験イテレーション (別名、経験生成イテレーション) は、ポリシーネットワークの重みの更新を実行する、各ポリシーイテレーション間の連続する経験エピソードのことです。各経験イテレーションの終わりに、収集されたエピソードが経験リプレイまたはバッファに追加されます。サイズは、トレーニング用のハイパーパラメータの 1 つに設定できます。ニューラルネットワークは、経験のランダムサンプルを使用して更新されます。

ポリシーイテレーション

ポリシーイテレーション (別名、ポリシー更新イテレーション) とは、勾配がきつくなる間、ポリシーニューラルネットワークの重みを更新するために、ランダムにサンプリングしたトレーニングデータを何回でも通過させることです。重みを更新するためのトレーニングデータの 1 回の学習は、エポックとも呼ばれます。

トレーニングジョブ

トレーニングジョブは、強化学習モデルをトレーニングし、推論の実行対象となるトレーニング済みモデル Artifact を作成するワークロードです。各トレーニングジョブには 2 つのサブプロセスがあります。

1. 現在のポリシーに従ってエージェントを起動する。エージェントはいくつかの [エピソード](#) で環境を調査し、トレーニングデータを作成します。このデータ生成自体は反復プロセスです。
2. 新しいトレーニングデータを適用して新しいポリシー勾配を計算する。ネットワークの重みを更新し、トレーニングを続行します。停止条件が満たされるまで、ステップ 1 を繰り返します。

各トレーニングジョブはトレーニング済みモデルを作成し、そのモデルアーティファクトを指定されたデータストアに出力します。

評価ジョブ

評価ジョブは、モデルのパフォーマンスをテストするワークロードです。トレーニングジョブの終了後、指定したメトリクスによりパフォーマンスが測定されます。標準的な AWS DeepRacer パフォーマンスメトリクスは、エージェントがトラックのラップの完走にかかる運転時間です。別のメトリクスとして、ラップの完走率もあります。

レースイベントの用語

AWS DeepRacer レースイベントでは、次の概念と用語を使用します。

リーグ/コンペティション

AWS DeepRacer League イベントのコンテキストでは、リーグと競争という用語は競争構造に関連しています。AWS は AWS DeepRacer League をスポンサーしています。つまり、AWS DeepRacer League を所有し、設計し、実行します。コンペティションには開始日と終了日があります。

Season

コンペティションは、毎年繰り返させる場合があります。これらはシーズン (2019 シーズンや 2020 シーズンなど) と呼ばれます。ルールはシーズンごとに変わる場合がありますが、同一シーズン内では通常同じです。AWS DeepRacer リーグの規約は、シーズンごとに異なる場合があります。

仮想サーキット

仮想サーキットとは、AWS DeepRacer League シーズン中に AWS DeepRacer コンソールで AWS 開催されるレースを指します。

イベント

ルールで定義されているとおり、イベントとは、ユーザーが参加できる AWS DeepRacer リーグの存在のことです。イベントには開始日と終了日があります。仮想サーキットイベントは通常 1 か月間続きます。1 つのシーズンに複数のイベントが存在する場合があります。イベント参加者のランク付け方法、勝利者の選択方法、その後の対応などに関するルールは、変更される可能性があります。

レースタイプ

すべてのレーサーはタイムトライアル (TT) レース、オブジェクト回避 (OA) レース、head-to-bot (H2B) レースでレースできます。レースタイプごとに、ラップ数、レーサーのランク付け方法を指定します。

ナショナルシーズン順位

ナショナルシーズン順位とは、その国の他のレーサー中でのレーサーのリーダーボードランキングを指します。すべてのレーサーは、毎月のバーチャルレースで自国の他のレーサーと競うことができます。

ナショナルシーズン順位

リージョナルシーズン順位とは、そのリージョンの他のレーサー中でのレーサーのリーダーボードランキングを指します。

ワールドチャンピオンシップ

AWS DeepRacer リーグの仮想サーキットマンスリーリーダーボードは、国と地域ごとに分かれています。各リージョンのトップレーサーは re AWS :Invent でワールドチャンピオンシップの資格を得ることができます。詳細については、「[用語と条件](#)」を参照してください。

AWS DeepRacer の仕組み

AWS DeepRacer 車両は、トラックを単独で運転したり、別の車両と対戦したりできる 1/18 スケールの自動運転車両です。前面カメラ、立体視カメラ、レーダー、LiDAR などの各種センサーを装備することができます。センサーは、車両が動作する環境に関するデータを収集します。さまざまなセンサーがさまざまなスケールでビューを提供します。

AWS DeepRacer では、強化学習を使用して、AWS DeepRacer 車両の自動運転を可能にします。これを実現するために、シミュレーションされたトラックがある仮想環境で、強化学習モデルをトレーニングおよび評価します。トレーニングの後、トレーニングしたモデル Artifact を AWS DeepRacer 車両にアップロードします。その後、車両が実際のトラックがある物理的環境で自動走行するように車両を設定できます。

特にこの分野の知識がない場合、強化学習モデルのトレーニングが困難な場合があります。AWS DeepRacer は必要なコンポーネントを統合し、タスクテンプレートのように分かりやすいウィザードを提供することで、プロセスを簡素化します。ただし、AWS DeepRacer に実装される強化学習トレーニングのベーシックを十分に理解すると役に立ちます。

トピック

- [AWS DeepRacer での強化学習](#)
- [AWS DeepRacer アクションスペースと報酬関数](#)
- [AWS DeepRacer トレーニングアルゴリズム](#)
- [AWS DeepRacer ソリューションのワークフロー](#)
- [Simulated-to-Real パフォーマンスギャップ](#)

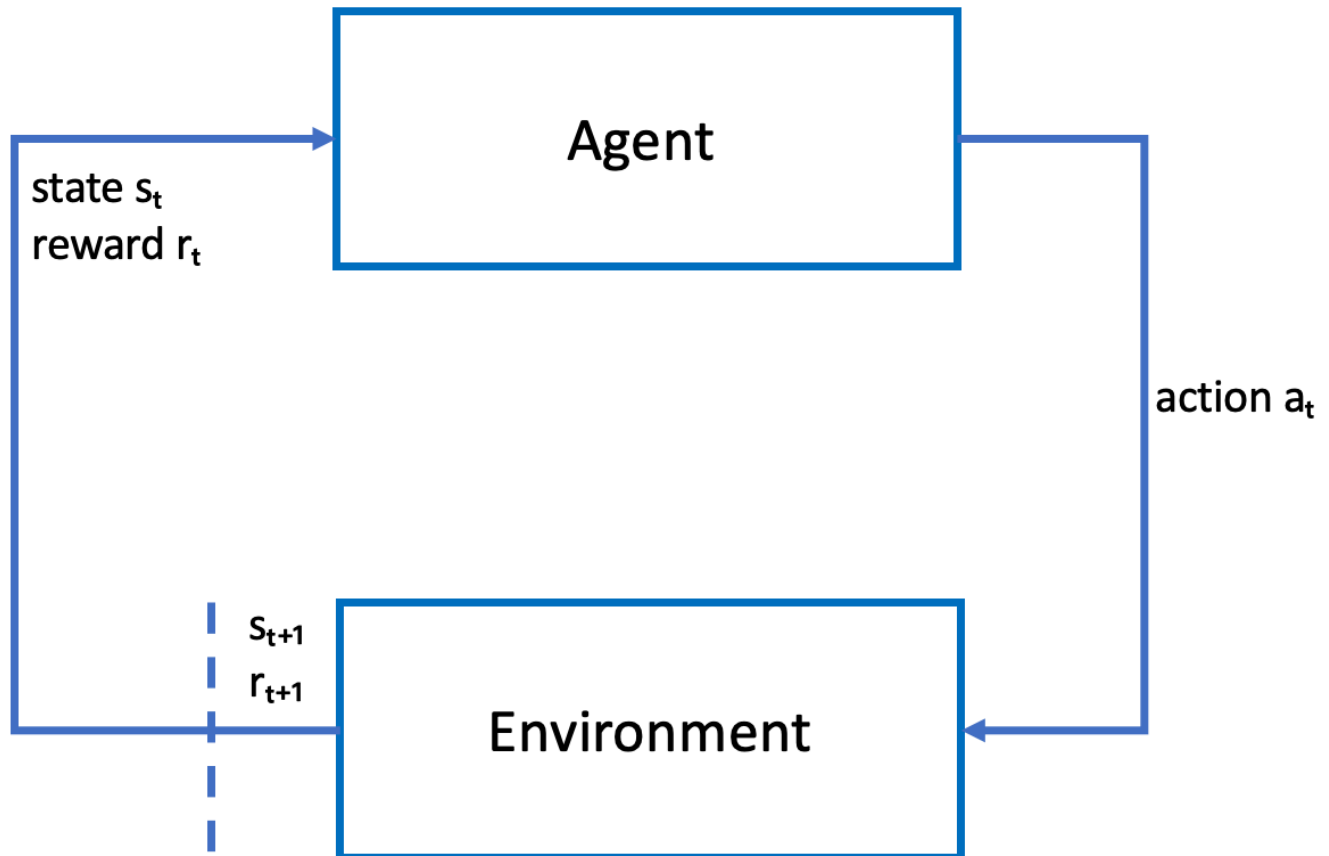
AWS DeepRacer での強化学習

強化学習では、エージェント物理または仮想の AWS DeepRacer 車両など、意図したといった、意思をもって目標を達成するためのという目的を持つことは、環境エージェントは、環境と通信してエージェントの総報酬を最大化します。エージェントは、一定の環境状態においてポリシーと呼ばれる戦略が導いたアクションを行い、新しい状態に達します。あらゆるアクションに対して、報酬が即座に関連付けられます。報酬は、アクションの優先度を決定するメジャーです。この即時報酬は、環境により返されるとみなされます。

AWS DeepRacer の強化学習の目的は、所定の環境における最適なポリシーを学習することです。学習とは、試行錯誤を繰り返すプロセスです。エージェントは初期アクションをランダムに実行して、

新しい状態に到達します。その後、エージェントは新しい状態からその次の状態に、ステップを繰り返します。そのうち、エージェントは最大の長期報酬につながるアクションを発見します。初期状態から最終状態までのエージェントのやりとりは、エピソードと呼ばれます。

次のスケッチはこの学習プロセスを示しています。



エージェントは、エージェントのポリシーを概算する関数を表すニューラルネットワークを具体化します。車両に搭載されたフロントカメラからの映像は環境の状態で、エージェントのアクションはエージェントの速度とステアリング角度で定義されます。

エージェントがトラック上を走行したままレースを終えると、エージェントは正の報酬を受け取り、トラックを外れると負の報酬を受け取ります。エピソードは、レーストラックのどこかでエージェントとともに開始し、エージェントが脱線するかラップを完走すると、終了します。

Note

厳密に言うと、環境の状態は問題に関係するすべてを指します。たとえば、トラック上の車両の位置とトラックの形状などです。車両のフロントに搭載されたカメラから供給される画像

では、環境状態のすべてはキャプチャされません。このため、環境は部分的に観察されているとみなされ、エージェントへのインプットは状態ではなく観測と呼ばれます。分かりやすいように、このドキュメントでは、状態と観測を同じ意味として使用します。

シミュレーションされた環境でエージェントをトレーニングすることには、次の利点があります。

- シミュレーションにより、エージェントがどの程度進歩を遂げたかを推定し、トラックから外れたタイミングを特定して、報酬を計算できます。
- シミュレーションにより、トレーナーは、物理環境のようにトラックから外れるたびに車両をリセットする、面倒な雑用から解放されます。
- シミュレーションはトレーニングを高速化できます。
- シミュレーションにより、たとえば、異なるトラック、背景、車両条件を選択するなど、環境の条件をより詳細に管理できます。

強化学習の代替は教師あり学習です。これは模倣学習とも呼ばれます。所定の環境から収集された ([image, action] タプルの) 既知のデータセットがあり、エージェントのトレーニングに使用されます。模倣学習を通して訓練されたモデルは、自動運転に適用することができます。カメラからの画像がトレーニングデータセットの画像と似ている場合にのみ効果があります。堅牢な走行には、トレーニングデータセットが包括的である必要があります。対照的に、強化学習はこのような大規模なラベリング作業を必要とせず、完全にシミュレーションで訓練することができます。強化学習はランダムなアクションで始まるため、エージェントは幅広い環境とトラック条件を学習します。このため、トレーニングしたモデルが堅牢になります。

AWS DeepRacer アクションスペースと報酬関数

アクションスペース

強化学習では、環境とやり取りする際にエージェントで利用できるすべての有効なアクションや選択肢の設定または選択肢をアクションスペースと呼ぶ環境と相互作用するためエージェントが利用できます。AWS DeepRacer コンソールでは、離散アクションスペースまたは連続アクションスペースのいずれかでエージェントをトレーニングできます。

離散アクションスペース

離散アクションスペースでは、有限集合内の各状態に対するエージェントの可能な作用がすべて表されます。AWS DeepRacer の場合、これは環境状況が段階的に変化するとともに、エージェントのニューラルネットワークがカメラおよび LiDAR センサー (オプション) からの入力に基づいて、自動

車の速度と方向を選択することを意味します。選択は、あらかじめ定義されたステアリングの角度とスロットル値の組み合わせに限定されます。

ターンに近づいている離散アクションスペースにある AWS DeepRacer 車両は、加速またはブレーキをかけて左、右、または直進のいずれかを選択できます。これらのアクションは、エージェントのオプション (0-9) のメニューを作成するステアリング角度と速度の組み合わせとして定義されます。例えば、0 は -30 度と 0.4 m/s を表すとして、1 は -30 度と 0.8 m/s、2 は -15 度と 0.4 m/s、そして 3 は -15 度と 0.8 m/s を表し、9 まで表すことができます。負の度数で車は右に曲がり、正の度数で車は左に曲がり、0 は車輪がまっすぐになります。

AWS DeepRacer のデフォルトの離散アクションスペースには、次のアクションが含まれます。

AWS DeepRacer のデフォルトの離散アクションスペース

アクションの数	ステアリング	速度
0	-30 度	0.4 m/s
1	-30 度	0.8 m/s
2	-15 度	0.4 m/s
3	-15 度	0.8 m/s
4	0 度	0.4 m/s
5	0 度	0.8 m/s
6	15 度	0.4 m/s
7	15 度	0.8 m/s
8	30 度	0.4 m/s
9	30 度	0.8 m/s

連続アクションスペース

連続アクションスペースにより、エージェントは各状態の値の範囲からアクションを選択できます。離散アクションスペースと同様に、これは徐々に異なる環境状況ごとに、エージェントのニューラル

ネットワークは、カメラと (オプションの) LiDAR センサーからの入力に基づいて、車の速度と方向を選択します。ただし、連続アクションスペースでは、エージェントがアクションを選択するオプションの範囲を定義できます。

この例では、連続アクションスペースでターンに近づく AWS DeepRacer 車両は、0.75 m/s から 4 m/s の速度を選択し、ステアリングの角度を-20 度から 20 度までの間で選択して左、右、または直進することができます。

離散 vs 連続

連続アクションスペースを使用する利点は、パフォーマンスを最適化するトラック上の特定ポイントでスピード/ステアリングアクションを動機付けするためのモデルをトレーニングする報酬機能が書き込みできることです。また、さまざまなアクションからピックアップすることで、スピードとステアリング値でスムーズな変化が可能になり、十分に訓練されたモデルであれば、実際の状況ではより良い結果が得られる可能性があります。

離散アクションスペースの設定では、エージェントの選択を事前定義されたアクションの有限数に制限することで、これらのアクションの影響を理解し、環境 (トラック、レース形式) と報酬関数に基づいてアクションを定義することができます。ただし、連続アクションスペースの設定では、エージェントはトレーニングを通じて提供される最小/最大範囲から最適な速度とステアリング値を選択することを学習します。

選択するモデルに対して値の範囲を設けて提供する方がよい選択肢だと考えられますが、エージェントは最適なアクションの選択方法を学ぶために長めにトレーニングする必要があります。成功させるためには報酬関数の定義にも依存します。

報酬関数

エージェントが環境を調べると、エージェントは値関数を学習します。value 関数を使用すると、エージェントが環境を観察した後、実行されたアクションがどの程度優良なのかを判断できます。value 関数は、AWS DeepRacer コンソールで記述した報酬関数を使用してアクションをスコアリングします。例えば、AWS DeepRacer コンソールの [follow the center line sample reward function] (センターラインのサンプル報酬関数の追跡) では、良いアクションはエージェントをトラックの中心近くに置き、悪いアクションより高くスコアリングしますが、そのエージェントはトラックの中心から離れることになります。

時間をかけて、value 関数により エージェントは総報酬を増加させるポリシーについて学ぶことができます。最適なポリシー (ベストポリシー) は、エージェントが環境の探索に費やす時間と、そのポリシーが経験を通じて学習したことを悪用または最大限に活用するのに費やす時間のバランスをとるものです。

センターラインに従う [\[AWS DeepRacer サンプル報酬関数の追跡\]](#) で、エージェントはまず環境を調査するためにランダムなアクションを実行します。つまり、トラックの中央に留まることはあまりありません。時間が経つにつれて、エージェントはどのアクションがセンターライン付近に留まるかを学習し始めますが、ランダムなアクションを続けてこれを行った場合、ラップ全体でトラックの中心付近に留まるための学習には時間がかかります。したがって、ポリシーが適切なアクションを学習し始めると、エージェントはランダムなアクションを実行するのではなく、これらのアクションを使用し始めます。ただし、適切なアクションを常に使用または悪用した場合、エージェントは環境を調査しなくなるため、新しい検出を行いません。このトレードオフは、RLの探索と悪用の問題と呼ばれます。

デフォルトのアクションスペースとサンプルの報酬関数を試してみてください。それらすべてを探索したら、独自の[カスタムアクションスペース](#)および[カスタムの報酬関数](#)を設計して知識を活かしてください。

AWS DeepRacer トレーニングアルゴリズム

PPO (Proximal Policy Optimization) と SAC (Soft Actor Critics)

アルゴリズム SAC と PPO は両方ともポリシーと value 関数を同時に学習しますが、戦略は次の 3 つの点で異なります。

PPO	SAC
離散アクションスペースと連続アクションスペースの両方で動作	連続アクションスペースで動作
オンポリシー	オフポリシー
エントロピー正則化を使用する	最大化目標にエントロピーを追加する

安定 vs データハングリー

環境を探索しながら、PPO アルゴリズムおよび SAC アルゴリズムのポリシーによって学習された情報はさまざまな方法で利用されます。PPO はオンポリシー学習を使用します。つまり、環境を探索する現在のポリシーによって行われた観測から、その value 関数を学習します。SAC は、オフポリシー学習を使用します。つまり、以前のポリシーの環境探索によって行われた観測が使用できます。オフポリシー学習とオンポリシー学習のトレードオフは、多くの場合、安定性とデータ効率性との比

較です。オンポリシーのアルゴリズムはより安定する傾向がありますが、データハングリーです。一方、オフポリシーのアルゴリズムはその逆になる傾向があります。

探索と活用との比較

探索と活用は、RL の重要な課題です。アルゴリズムは、より高い累積報酬を達成するために以前の経験からの既知の情報を利用する必要がありますが、将来的に最適なポリシーを見つける際に使用できる新しい経験を得るためにも探索する必要があります。ポリシーが何度も反復してトレーニングされ、環境についての学習がさらに進むため、特定の観測に対するアクションの選択がより明確になります。ただし、ポリシーが十分に検討できない場合、最適でない場合でも、すでに学習した情報に固執する可能性があります。PPO アルゴリズムは、エージェントを局所最適に収束させないようにエントロピー正則化を用いることで探索を促進します。SAC アルゴリズムは、その最大化目標にエントロピーを加えることによって、探索と活用の間で例外的なバランスを取ります。

エントロピー

この文脈において、「エントロピー」はポリシーにおける不確実性の尺度であるため、あるポリシーが特定の状態に対するアクションを選択する際にどの程度の自信があるかという尺度として解釈できます。エントロピーが低いポリシーは、アクションを選択する自信が高いのに対し、エントロピーが高いポリシーはどのアクションを選択するか不明確です。

SAC アルゴリズムのエントロピー最大化戦略は、PPO アルゴリズムのエントロピーを正則化するものとして使用する場合と同様の利点があります。PPO と同様に、より広い探索を推進し、エントロピーのより高いアクションを選択するようエージェントに動機づけを与えることによって、悪い局所最適へ収束しないようにします。エントロピーの規則とは異なり、エントロピーの最大化には独自の利点があります。また、不確実な動作を選択するポリシーはあきらめる傾向がありますが、SAC アルゴリズムが PPO よりもデータ効率が高い傾向にあるもう一つの理由がこれです。

SAC アルファハイパーパラメーターを使用して、SAC のエントロピーの量を調整します。SAC アルファエントロピーの最大値 (1.0) は、探索に適しています。最小値 (0.0) は、標準的な RL の目的を回復し、探索にインセンティブを与えるエントロピーボーナスを中和します。実験を始めるのに適切な SAC アルファ値は 0.5 です。モデルの反復処理に応じて調整します。

PPO アルゴリズムと SAC アルゴリズムの両方でハイパーパラメーターを試して、さまざまなアクションスペースで探索してください。

AWS DeepRacer ソリューションのワークフロー

AWS DeepRacer モデルのトレーニングには、以下の一般的なタスクが含まれます。

1. AWS DeepRacer サービスは、仮想トラック、車両に対応するエージェント、背景でシミュレーションを初期化します。エージェントはポリシーニューラルネットワークを具体化します。これは、[PPO アルゴリズム](#)で定義されたとおりにハイパーパラメータを使って調整できます。
2. エージェントは、所定の状態 (フロントカメラからの画像で表す) に基づいて動作します (ステアリング角度と速度で指定)。
3. シミュレーションされた環境は、エージェントのアクションに基づいてエージェントの位置を更新し、報酬と更新されたカメラ画像を返します。状態、アクション、報酬、新しい状態の形で収集された経験は、ニューラルネットワークを定期的に更新するために使用されます。更新されたネットワークモデルは、より多くの体験を生み出すために使用されます。
4. エージェントから見た一人称ビューで、シミュレーションされたトラックに沿って進行中のトレーニングを監視できます。エピソードごとの報酬、損失関数値、ポリシーのエントロピーなどのメトリックを表示できます。トレーニングが進むにつれて、CPU またはメモリの使用率も表示できます。さらに、分析とデバッグのため、詳細なログが記録されます。
5. AWS DeepRacer サービスでは、ニューラルネットワークモデルを定期的に永続的ストレージに保存します。
6. 時間制限に基づいてトレーニングは停止します。
7. シミュレーターでトレーニングされたモデルを評価できます。評価するには、選択したトラックを選択した回数走行するタイムトライアルに、トレーニング済みモデルを提出します。

モデルが正常にトレーニングされ評価された後、モデルは物理エージェント (AWS DeepRacer 車両) にアップロードされます。このプロセスには、以下のステップが含まれます。

1. 永続的ストレージからトレーニング済みモデルをダウンロードします (Amazon S3 バケット)。
2. 車両のデバイスコントロールコンソールを使用して、デバイスにトレーニング済みモデルをアップロードします。コンソールを使用して、シミュレーションされたアクションスペースを物理的なアクションスペースにマッピングするために車両をキャリブレートします。またコンソールを使用して、スロットルパリティを確認したり、フロントカメラフィードを表示してモデルを推論エンジンにロードしたり、実際のトラックで運転している車両を見ることもできます。

車両のデバイスコントロールコンソールは、車両のコンピューティングモジュールにホストされている Web サーバーです。接続された Wi-Fi ネットワークとコンピュータまたはモバイルデバイスの Web ブラウザを使用して、車両の IP アドレスからコンソールにアクセスできます。

3. さまざまな照明、バッテリーレベル、表面の質感や色の条件下で、車両の運転を経験します。

モデルの制限やトレーニング不足のために、物理的環境でのデバイスのパフォーマンスがシミュレーション環境でのパフォーマンスと一致しない場合があります。この現象は sim2real パフォー

パフォーマンスギャップと呼ばれます。ギャップを減らすには、「[the section called “Simulated-to-Real パフォーマンスギャップ”](#)」を参照してください。

Simulated-to-Real パフォーマンスギャップ

シミュレーションでは実世界のすべての側面を正確に把握することはできないため、シミュレーションでトレーニングされたモデルが実世界ではうまく機能しない場合があります。このような不一致は、多くの場合 simulated-to-real (sim2real) パフォーマンスギャップと呼ばれます。

AWS DeepRacer では sim2real パフォーマンスギャップを最小化する取り組みが実施されてきました。たとえば、シミュレーションされたエージェントは、毎秒約 10 のアクションを実行するようにプログラムされています。これは、AWS DeepRacer 車両が推論する頻度と一致し、毎秒約 10 推論です。別の例として、トレーニング中の各エピソードの始めに、エージェントの位置はランダム化されます。これにより、エージェントがトラックのすべての部分を均等に学習する可能性が最大になります。

real2sim パフォーマンスギャップを減らすため、必ずシミュレーションされたトラックと実際のトラックの両方に、同じまたは類似の色、形状、ディメンションを使用してください。視覚的に邪魔になるものを減らすには、実際のトラックの周囲にバリケードを使用します。また、トレーニングで使用されるアクションスペースが現実の世界と一致するように、デバイスの速度とステアリング角度の範囲を慎重にキャリブレートしてください。トレーニングで使用されているものとは異なるシミュレーショントラックでモデルのパフォーマンスを評価すると、real2real パフォーマンスのギャップの程度を示すことができます。

AWS DeepRacer モデルをトレーニングする場合の sim2real ギャップの削減方法についての詳細は、[the section called “実環境のトレーニングを最適化する”](#) を参照してください。

AWS DeepRacer の使用開始

AWS DeepRacer を使い始めるために、まずは AWS DeepRacer コンソールを使って、自動運転の要件に適したセンサーを持つエージェントを設定し、指定したセンサーを持つエージェントに対して強化学習モデルをトレーニングして、学習したモデルを評価し、モデルの品質を確認するステップを見ていきましょう。モデルをトレーニングしたら、それを繰り返し処理してレースに送信できます。

トピック

- [最初の AWS DeepRacer モデルのトレーニング](#)
- [シミュレーションでの AWS DeepRacer モデルの評価](#)

最初の AWS DeepRacer モデルのトレーニング

このチュートリアルでは、AWS DeepRacer コンソールを使って最初のモデルをトレーニングする方法を紹介します。

AWS DeepRacer コンソールを使用して強化学習モデルをトレーニングします。

AWS DeepRacer コンソールの [モデルの作成] ボタンをクリックし、モデルトレーニングの旅をスタートします。

強化学習モデルをトレーニングするには

1. AWS DeepRacer を初めて使用する場合は、サービスのランディングページから [モデルの作成] を選択するか、メインのナビゲーションペインの見出しにある [強化学習] で、[スタートする] を選択します。
2. [強化学習をスタートする] ページの [ステップ 2: モデルとレースを作成する] で、[モデルの作成] を選択します。

または、メインのナビゲーションペインの見出しにある [強化学習] で、[モデル] を選択します。[モデル] ページで、[モデルの作成] を選択します。

モデル名と環境を指定します。

モデルに名前を付けて、自分に合ったシミュレーショントラックを選択する方法を学びましょう。

モデル名と環境を指定

1. [モデルの作成] ページの、[トレーニングの詳細] で、モデル名を入力します。
2. 必要に応じて、トレーニング ジョブの説明を追加します。
3. オプションタグの追加の詳細については、「[Tagging](#)」を参照してください。
4. [環境シミュレーション] で、AWS DeepRacer エージェントのトレーニング環境として機能するトラックを選択します。[トラック方向] で、時計回りまたは反時計回りを選択します。続いて、[次へ] を選択します。

最初の実行では、シンプルな形でターンの滑らかなトラックを選択します。後の反復では、モデルを徐々に改善するために、より複雑なトラックを選択できます。特定のレースイベントのモデルをトレーニングするには、そのイベントのトラックに最も近いトラックを選択します。

5. ページの一番下の [次へ] を選択します。

レースタイプとトレーニングアルゴリズムを選択します。

AWS DeepRacer コンソールには、3 つのレースタイプと 2 つのトレーニングアルゴリズムがあり、そこから選択できます。あなたのスキルレベルとトレーニングの目標に適したものを把握しましょう。

レースタイプとトレーニングアルゴリズムを選択する方法

1. 「モデルの作成」ページの、[レースタイプ] で、[タイムトライアル]、[オブジェクトの回避]、または [Head-to-bot] を選択します。

最初の走行では、[タイムトライアル] を選択することをお勧めします。このレースタイプで、エージェントのセンサー構成を最適化するためのガイダンスについては、「[the section called “タイムトライアル向けのレーニングのカスタマイズ”](#)」を参照してください。

2. その後の走行では、[オブジェクト回避] を選択して、選択したトラックに沿って固定した位置、またはランダムな位置に配置された静止障害物を避けるオプションを選ぶことができます。詳細については、「[the section called “オブジェクト回避レース向けのトレーニングのカスタマイズ”](#)」を参照してください。
 - a. トレーニングシミュレーションの各エピソードの開始時に、[固定位置] を選択して、トラックの 2 つのレーンの固定位置にボックスを生成するか、[ランダムな位置] を選択して、2 つのレーンにランダムに分散されるオブジェクトを生成します。
 - b. 次に、[トラック上のオブジェクトの数] の値を選択します。

- c. 「固定位置」を選択した場合は、トラック上の各オブジェクトの位置を調整できません。[レーンの配置] では、内側レーンと外側レーンのどちらかを選択します。デフォルトでは、オブジェクトはトラック全体に均等に分散されます。オブジェクトの開始位置と終了位置の間の距離を変更するには、「開始点と終了点の間の位置 (%)」フィールドに、その距離のパーセンテージを 7~90 の間で入力します。
3. より野心的な走行を行うには、[Head-to-bot レース] を選択して、一定の速度で移動するボット車両 (最大4台) と対戦するオプションもあります。詳細については、「[the section called "head-to-bot レース向けのトレーニングのカスタマイズ"](#)」を参照してください。
 - a. [ボット車両の数を選ぶ] で、エージェントにトレーニングさせたいボット車両の数を選択します。
 - b. 次に、ボット車両がトラック内を移動する速度を、ミリメートル/秒で選択します。
 - c. [レーンの変更を有効にする] ボックスをチェックして、ボット車両に 1~5 秒ごとにランダムにレーンを変更する機能を与えるオプションもあります。
 4. [トレーニングアルゴリズムとハイパーパラメータ] で、[Soft Actor Critic (SAC)] または [Proximal Policy Optimization (PPO)] アルゴリズムを選択します。AWS DeepRacer コンソールでは、SAC モデルを連続アクションスペースでトレーニングする必要があります。PPO モデルは、連続または離散アクションスペースのいずれかでトレーニングさせることができます。
 5. [トレーニングアルゴリズムとハイパーパラメータ] で、デフォルトのハイパーパラメータ値をそのまま使用します。

後で、トレーニングのパフォーマンスを向上させるには [ハイパーパラメータ] を展開し、デフォルトのハイパーパラメータ値を次のように変更します。

- a. [勾配降下バッチサイズ] には、[利用可能なオプション](#)を選択します。
- b. [エポック数] には [有効な値](#) を設定してください。
- c. [学習レート] には、[有効な値](#) を設定してください。
- d. [SAC アルファ値] (SAC アルゴリズムのみ)、[有効な値](#) を設定します。
- e. [エントロピー] には、[有効な値](#) を設定してください。
- f. [割引係数] には、[有効な値](#) を設定してください。
- g. [損失タイプ] では、[利用可能なオプション](#)を選択してください。
- h. [各ポリシー更新反復の間エクスペリエンスエピソードの数] には、[有効な値](#) を設定してください。

ハイパーパラメータの詳細については、「[ハイパーパラメータを体系的に調整する](#)」を参照してください。

6. [次へ] を選択します。

アクションスペースの定義

[アクションスペースの定義] ページで、Soft Actor Critic (SAC) アルゴリズムを使用してトレーニングすることを選択した場合、デフォルトのアクションスペースは連続アクションスペースとなります。Proximal Policy Optimization (PPO) アルゴリズムを使用してトレーニングすることを選択した場合、[連続アクションスペース] と [離散アクションスペース] のいずれかを選択します。各アクションスペースとアルゴリズムがエージェントのトレーニング エクスペリエンスをどのように形作るかについての詳細はこちら「[the section called “アクションスペースと報酬関数”](#)」をご覧ください。

連続アクション空間 (SAC または PPO アルゴリズム) を定義するには

1. [連続アクションスペースの定義] で、[左側のステアリング角度範囲] と [右側のステアリング角度範囲] の度数を選択します。

各ステアリング角度の範囲に異なる度数を入力し、[動的セクターグラフ] で視覚化された、その範囲の変化を確認します。

Define continuous action space [Info](#)

In a continuous action space setting, the agent learns to pick the optimal speed and steering values from the min/max bounds you provide through training. Providing a range of values for the model to pick from seems to be the better option but the agent has to train longer to learn to choose the optimal actions.

Steering angle

The steering angle determines the range of steering angles in which the front wheels of your agent can turn.

Left steering angle range

degrees

Values are between 0 and 30.

Right steering angle range

degrees

Values are between -30 and 0.

Speed

The speed determines how fast your agent can drive.

Min/max speed defines the range of speeds available to the agent while training.

Minimum speed

m/s

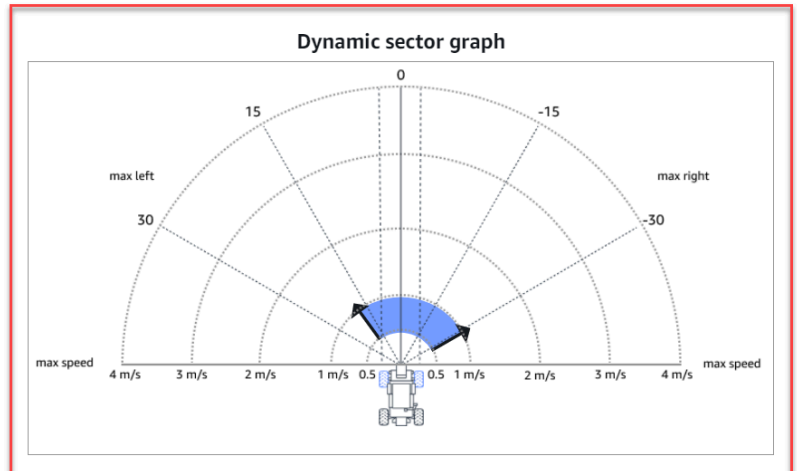
Values are between 0.5 and 4.

Maximum speed

m/s

Values are between 0.5 and 4.

[Reset to default values](#)



Cancel

Previous

Next

2. [速度] で、エージェントの最小速度と最高速度をミリメートル/秒で入力します。

変更がどのように [動的セクターグラフ] に反映されるかに注目してください。

3. 必要に応じて、[デフォルト値にリセット] を選択して、不要な値をクリアします。実験や学習のために、グラフでさまざまな値を試してみることをお勧めします。
4. [次へ] を選択します。

離散アクション空間を定義するには (PPO アルゴリズムのみ)

1. ドロップダウンリストから、[ステアリング角度の粒度] の値を選択します。
2. エージェントの [最大ステアリング角度] 度数の値を 1 ~ 30 の範囲で選択します。

3. ドロップダウンリストから [速度の粒度] の値を選択します。
4. エージェントの [最大速度] を 0.1~4 の範囲の値から、ミリメートル/秒で選択します。
5. [アクションリスト] にデフォルトのアクション設定を使用するか、オプションで [アドバンスト設定] をオンに切り替えて、設定をチューニングします。[戻る] を選択する、または値を調整した後で [アドバンスト設定] をオフに切り替えると、変更内容が失われます。

Select action space [Info](#)

Action spaces

Continuous action space
A continuous action space allows the agent to select an action from a range of values for each state.

Discrete action space
A discrete action space represents all of the agent's possible actions for each state in a finite set.

Define discrete action space [Info](#)

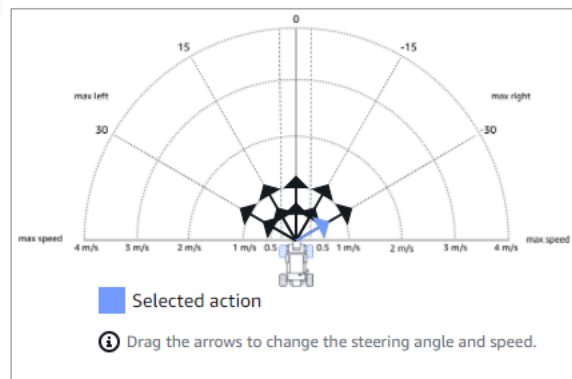
Action list

 Advanced configuration

Action	Steering angle Choose between -30 and 30	Speed Choose between 0.1 and 4
0	-30 degrees	0.5 m/s
1	-30 degrees	1 m/s
2	-15 degrees	0.5 m/s
3	-15 degrees	1 m/s
4	0 degrees	0.5 m/s
5	0 degrees	1 m/s
6	15 degrees	0.5 m/s
7	15 degrees	1 m/s
8	30 degrees	0.5 m/s
9	30 degrees	1 m/s

A new action will be added with the values of the last action in the table.
You can add up to 11 more actions.

Radial polar graph



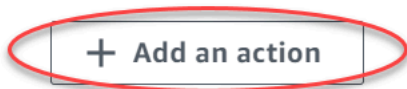
Cancel

Previous

Next

- [ステアリング角度] 列に -30 から 30 の範囲の値を入力します。
- 最大 9 つのアクションに対して、[速度] 列に 0.1~4 の値をミリメートル/秒で入力します。
- 必要に応じて、[アクションを追加する] を選択し、アクションリストの行数を増やします。

5	<input type="text" value="0"/>	degrees	<input type="text" value="1"/>	m/s ✕
6	<input type="text" value="0"/>	degrees	<input type="text" value="1"/>	m/s ✕
7	<input type="text" value="0"/>	degrees	<input type="text" value="1"/>	m/s ✕
8	<input type="text" value="0"/>	degrees	<input type="text" value="1"/>	m/s ✕
9	<input type="text" value="0"/>	degrees	<input type="text" value="1"/>	m/s ✕
10	<input type="text" value="0"/>	degrees	<input type="text" value="1"/>	m/s ✕
11	<input type="text" value="0"/>	degrees	<input type="text" value="1"/>	m/s ✕
12	<input type="text" value="0"/>	degrees	<input type="text" value="1"/>	m/s ✕
13	<input type="text" value="0"/>	degrees	<input type="text" value="1"/>	m/s ✕
14	<input type="text" value="0"/>	degrees	<input type="text" value="1"/>	m/s ✕
15	<input type="text" value="0"/>	degrees	<input type="text" value="1"/>	m/s ✕



A new action will be added with the values of the last action in the table. You can add up to 5 more actions.

d. 必要に応じて、行にある ✕ を選択して削除します。

6. [次へ] を選択します。

バーチャルカーを選択

バーチャルカーで使用をスタートする方法 毎月オープンディビジョンに出場して、新しいカスタムカー、ペイントジョブ、改造をゲットしましょう。

バーチャルカーを選択する方法

1. [車両シェルとセンサー構成の選択] ページで、自分のレースタイプおよびアクションスペースと互換性のあるシェルを選択します。一致する車がガレージにない場合は、メイン ナビゲーションペインの見出しにある [強化学習] で、[ガレージ] を選択して作成します。

[タイムトライアル] トレーニングに必要なのは、オリジナルの DeepRacer にある、デフォルトのセンサー構成と一眼レフカメラだけです。他のすべてのシェルとセンサー構成は、アクションスペースが一致している場合に限り機能するようになっています。詳細については、「[the section called “タイムトライアル向けのレーニングのカスタマイズ”](#)」を参照してください。

ステレオカメラは オブジェクトの回避 トレーニングで役に立ちますが、固定位置にある静止した障害物を回避するためにはシングルカメラも使用することができます。LiDAR センサーは、オプションです。「[the section called “アクションスペースと報酬関数”](#)」を参照してください。

Head-to-bot トレーニングでは、シングルカメラまたはステレオカメラに加え、他の動いている車両を追い越しながら死角を検出して回避するのに、LiDAR ユニットが最適です。詳細については、[the section called “head-to-bot レース向けのトレーニングのカスタマイズ”](#) を参照してください。

2. [次へ] を選択します。

報酬機能をカスタマイズします。

報酬関数は強化学習の中核です。それを利用して、トラック (環境) を探索するときに車 (エージェント) が特定の行動を起こすようにインセンティブを与える方法を学びます。ペットの特定の行動を奨励および阻止するように、このツールを使用して、車ができるだけ早くラップを終えるように促し、トラックから外れたり、物体に衝突したりしないようにすることができます。

報酬機能をカスタマイズする方法

1. [モデルの作成] ページの [報酬関数] で、最初のモデルのデフォルトの報酬関数の例をそのまま使用します。

Reward function [Info](#)

The reward function describes immediate feedback (as a score for reward or penalty) when the vehicle takes an action to move from a given position on the track to a new position. Its purpose is to encourage the vehicle to make moves along the track to reach its destination quickly. The model training process will attempt to find a policy which maximizes the average total reward the vehicle experiences.

Code editor

Reward function examples

Reset

Validate

```

1 def reward_function(params):
2     """
3     Example of rewarding the agent to follow center line
4     """
5
6     # Read input parameters
7     track_width = params['track_width']
8     distance_from_center = params['distance_from_center']
9
10    # Calculate 3 markers that are at varying distances away from the center line
11    marker_1 = 0.1 * track_width
12    marker_2 = 0.25 * track_width
13    marker_3 = 0.5 * track_width
14
15    # Give higher reward if the car is closer to center line and vice versa
16    if distance_from_center <= marker_1:
17        reward = 1.0
18    elif distance_from_center <= marker_2:
19        reward = 0.5
20    elif distance_from_center <= marker_3:
21        reward = 0.1
22    else:
23        reward = 1e-3 # likely crashed/ close to off track
24
25    return float(reward)

```

後で、[報酬関数の例] を選択して別の関数の例を選択し、[コードを使用] を選択して、選択した報酬関数を受け入れることができます。

4 つのサンプル関数からスタートすることができます。これらは、トラックの中央を走行する方法 (デフォルト)、トラックの境界線内にエージェントを維持する方法、ジグザグ運転を防止する方法、静止障害物やその他の走行車両への衝突を回避する方法を説明しています。

報酬関数の詳細については、「[the section called “報酬関数リファレンス”](#)」を参照してください。

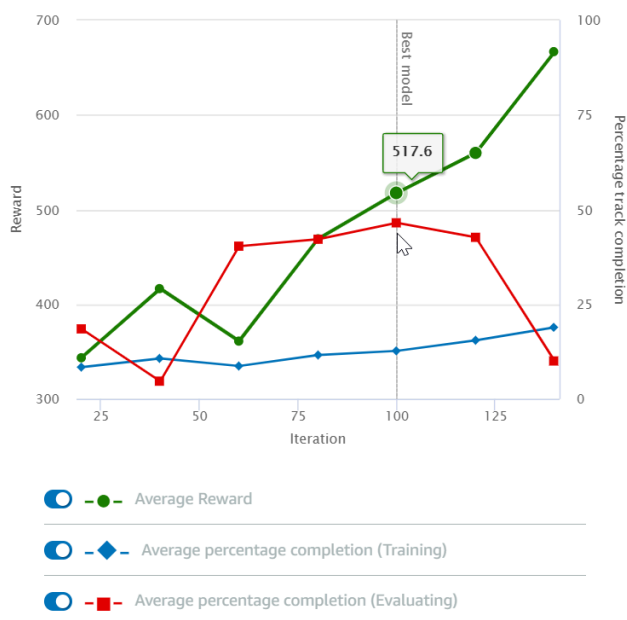
2. [モデルの作成] ページの [停止条件] で、既定の [最大時間] 値をそのままにしておくか、新しい値を設定してトレーニングジョブを終了し、長時間の (また暴走の可能性のある) トレーニングジョブを防止します。

トレーニングの初期段階で実験を行うときは、このパラメータの小さい値からスタートし、徐々にトレーニングを長くしていきます。

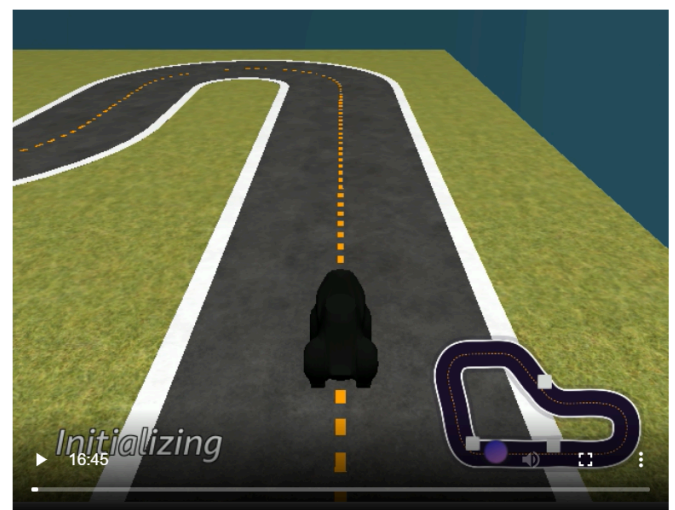
3. [AWS DeepRacer に自動的に送信する]にある、[トレーニング完了後にこのモデルを AWS DeepRacer に自動的に送信し、賞品を獲得するチャンスを得る]は、デフォルトでチェックされています。オプションとして、チェックマークを選択してモデルの入力をオプトアウトできます。
4. [リーグ要件]で[居住国]を選択し、チェックボックスをオンにして利用規約に同意します。
5. [モデルの作成]を選択して、モデルの作成とトレーニング ジョブのインスタンスのプロビジョニングを開始します。
6. 送信後、トレーニング ジョブが初期化されていることを確認してから実行します。

初期化プロセスでは、ステータスが [初期化中] から [進行中] になるまでに約 6 分かかります。

7. [報酬グラフ]と[シミュレーション ビデオ ストリーム]で、トレーニング ジョブの進捗状況を確認します。[報酬グラフ]の横にある更新ボタンを定期的を選択すると、トレーニング ジョブが完了するまで [報酬グラフ]を更新できます。

Reward graph [Info](#)

Simulation video stream



トレーニングジョブは AWS クラウドで実行されるため、AWS DeepRacer コンソールを開いたままにする必要はありません。ジョブの進行中はいつでもコンソールに戻ってモデルをチェックできます。

[シミュレーション ビデオ ストリーム] ウィンドウ、または [報酬グラフ] の表示が反応しなくなった場合は、ブラウザ ページを更新してトレーニングの進行状況を更新します。

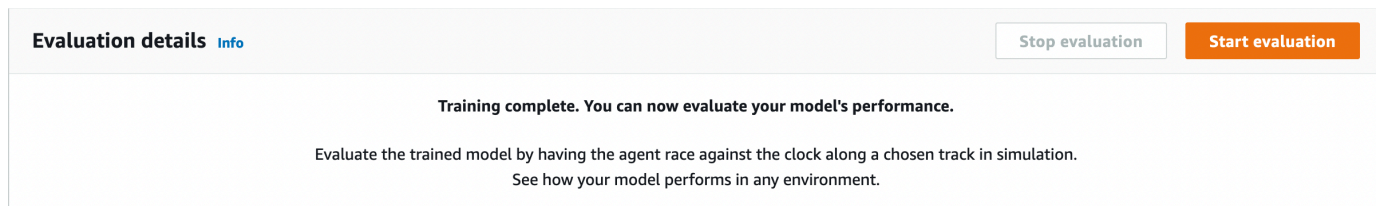
シミュレーションでの AWS DeepRacer モデルの評価

トレーニングジョブの完了後、トレーニング済みモデルを評価して収束動作を評価する必要があります。評価は、選択したトラックで何回かのトライアルを完了し、トレーニング済みモデルによって推定される可能性のあるアクションに従ってエージェントをトラック上で移動させることによって行われます。パフォーマンスメトリクスには、トラックの完了率と、各トラックの開始から終了またはオフトラックになるまでの時間が含まれます。

トレーニング済みモデルを評価するには、AWS DeepRacer コンソールを使用します。そのためには、このトピックのステップを実行します。

AWS DeepRacer コンソールでトレーニング済みモデルを評価する方法

1. <https://console.aws.amazon.com/deepracer> で AWS DeepRacer コンソールを開きます。
2. メインナビゲーションペインで [モデル] を選択し、トレーニングしたモデルを [モデル] リストから選択して、モデルの詳細ページを開きます。
3. [評価] タブを選択します。
4. [評価] で [評価の開始] を選択します。



トレーニングジョブのステータスが [完了] に変わった後、またはトレーニングジョブが完了していない場合は、モデルのステータスが [準備完了] に変わった後で評価を開始できます。

モデルは、トレーニングジョブが完了したときに準備完了状態になります。トレーニングが完了しなかった場合でも、失敗した時点までトレーニングされていれば、モデルは [準備完了] 状態になることができます。

5. [モデルの評価] ページの [レースタイプ] で、モデルをトレーニングするために選択したレースタイプを選択します。

評価のために、トレーニングで使用したレースタイプとは異なるレースタイプを選ぶことができます。たとえば、head-to-bot レース向けにモデルをトレーニングして、タイムトライアルで評価することができます。一般に、トレーニングレースタイプが評価レースタイプと異なる場合、モデルは適切に一般化する必要があります。最初の走行では、評価とトレーニングの両方で同じレースタイプを使用する必要があります。

6. [モデルの評価] ページの [評価基準] で、モデルの評価に使用するトライアルの回数を選択します。

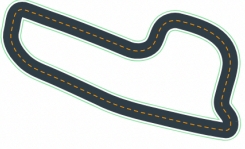
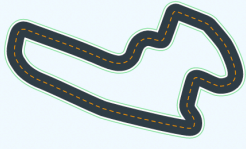

Evaluate criteria [Info](#)

Choose the track you want to use to evaluate your model.

Choose number of trials to evaluate your model

3 trials

Simulated environment emulates a track to train your model.

<p><input type="radio"/> Jennens Family Speedway</p> <p>The Jennens Family Speedway (49.56 m) is named in honor of the first ever racing family and 2021 re:Invent finalists James "JJ" and Timothy "Flatearth" Jennens. This track features two blistering fast drag strips right into unforgiving 90 degree sweeping turns that can spin out even the most skilled developers.</p>  <p>Open division qualifier</p>	<p><input checked="" type="radio"/> Jennens Super Speedway</p> <p>The Jennens Super Speedway (62.07 m) is named in honor of the first ever racing family and 2021 re:Invent finalists James "JJ" and Timothy "Flatearth" Jennens. Pros this month will need to navigate the technical section without off tracks, and apply top speeds on the drag strips to climb the leaderboard.</p>  <p>Pro division qualifier</p>	<p><input type="radio"/> 2022 re:Invent Championship</p> <p>Get ready to rev your engines on the official 2022 re:Invent Championship track! This is an intensely difficult track (35.87 m) featuring a technical chicane section that will challenge even the most skilled developers.</p>  <p>re:Invent track</p>
---	---	---

[View more race track options](#)

一般的には、[モデルのトレーニング](#)で使用したものと類似のトラックを選択します。モデルを評価するために任意のトラックを選択できますが、トレーニングで使用されるものともよく似たトラックで最高のパフォーマンスが期待できます。

モデルが適切に一般化するかどうかを確認するには、トレーニングで使用したものと異なる評価トラックを選択します。

7. 最初のモデルの [モデルの評価] ページにある [バーチャルレースの提出] で、[評価後にモデルを提出] オプションをオフにします。後で、レースイベントに参加する場合は、このオプションをオンにしておきます。

Virtual race submission

Virtual races [Info](#)

Congratulations training your model, now see how your model stacks up. Submit your model to participate in the virtual race. Your model will be ranked based on the average time it takes to complete a lap on the race track. Your results will be displayed on the leaderboard. Win prizes, no fees or costs for entering the virtual league and unlimited race submissions.

Submit model after evaluation

Win prizes, no fees or costs for submitting a model to the virtual league.

8. [モデルの評価] ページで、[評価の開始] を選択して、評価ジョブの作成と初期化を開始します。

この初期化プロセスは完了に 3 分かかります。

9. 評価が進むにつれて、トライアルの時間やトラック完了率などの評価結果が、各トライアルの後に [評価の詳細] に表示されます。[シミュレーションビデオストリーム] ウィンドウでは、選択したトラックでのエージェントのパフォーマンスを観察できます。

評価ジョブは、完了する前に停止できます。評価ジョブを停止するには、[評価] カードの右上隅にある [評価を停止] を選択し、評価の停止を確定します。

10. 評価ジョブが完了したら、[評価結果] の下にあるすべてのトライアルについて、パフォーマンスメトリクスを調べます。付属のシミュレーションビデオストリームは使用できなくなりました。

「評価セレクター」には、モデルの評価履歴が表示されます。特定の評価の詳細を表示するには、[評価セレクター] リストから評価を選択し、[評価セレクター] カードの右上隅にある [負荷の評価] を選択します。

Evaluation selector (1/1) Load evaluation

Find evaluations < 1 > ⚙

Name	Evaluation date	Lap time	Track	Race type
Doc-Example	11/14/2022	00:54.858	Jennens Super Speedway	Time trial

Doc-Example evaluation details [Info](#) Download logs Stop evaluation Start new evaluation

Simulation video stream



Evaluation results

Trial	Time (MM:SS.mmm)	Trial results (% track completed)	Status
1	00:54.858	100%	Lap com
2	01:03.531	100%	Lap com
3	01:03.132	100%	Lap com

この特定の評価ジョブでは、トレーニングを受けたモデルがトライアルを完了すると、大幅な場外時間ペナルティが発生します。最初の走行としては、これは珍しいことではありません。考えられる理由としては、トレーニングが収束せず、トレーニングにさらに時間が必要である、工一

ジェントに反応する余地を与えるためにアクションスペースを拡大する必要がある、またはさまざまな環境を処理するために報酬関数を更新する必要がある、などが挙げられます。

トレーニング済みのモデルのクローンを作成し、報酬関数を変更し、ハイパーパラメータを調整してから、総報酬が収束してパフォーマンス指標が向上するまでプロセスを繰り返すことで、モデルを継続的に改善できます。トレーニングを向上する方法の詳細については、「[モデルのトレーニングと評価](#)」を参照してください。

完全にトレーニングされたモデルを AWS DeepRacer デバイスに転送して、実環境で走行させるには、モデルのアーティファクトをダウンロードする必要があります。これを行うには、モデルの詳細ページで [モデルのダウンロード] を選択します。AWS DeepRacer の物理デバイスが新しいセンサーをサポートしておらず、モデルが新しいセンサータイプでトレーニングされている場合、実環境で AWS DeepRacer デバイスにモデルを使用すると、エラーメッセージが表示されます。物理デバイスを使用した AWS DeepRacer モデルのテストに関する詳細については、「[車両を運転する](#)」を参照してください。

AWS DeepRacer League レーシングイベントや AWS DeepRacer のコミュニティレースで指定されたトラックと同一または類似のトラックでモデルをトレーニングしたら、AWS DeepRacer コンソールで仮想レースにモデルを送信することが可能になります。これを行うには、メインナビゲーションペインで [AWS 公式 DeepRacer 仮想サーキット] または [コミュニティレース] に従ってください。詳細については、「[レースに参加する](#)」を参照してください。

障害物回避や head-to-bot レースのモデルをトレーニングするには、モデルと物理デバイスに新しいセンサーを追加する必要がある場合があります。詳細については、「[the section called “レースの種類を理解とセンサーの有効化”](#)」を参照してください。

AWS DeepRacer モデルのトレーニングと評価

AWS DeepRacer 車両がトラックで自動走行するとき、前面にマウントされたカメラで環境状態をキャプチャし、観察データに応じてアクションを実行します。AWS DeepRacer モデルは、観察データとアクションを想定される報酬に マッピングする関数です。モデルをトレーニングすることは、想定される報酬を最大化する関数を見つける、または学習することであり、最適化されたモデルによって車両が最初から最後までトラックで走行するためにどのようなアクション (速度とステアリング角度のペア) を取れるかを規定します。

実際には、関数はニューラルネットワークで表され、ネットワークのトレーニングには、一連の観察された環境状態と対応する車両のアクションに与えられた最適なネットワークの重み付けを見つけることが関係しています。最適性の基盤となる基準は、車両が交通事故または違反を発生させることなく正しく効果的な動きをするように勧める、モデルの報酬関数によって記述されます。シンプルな報酬関数は、車両がトラックにある場合は 0、トラックから外れた場合は -1、ゴールに到達した場合は +1 の報酬を返すようにすることができます。この報酬関数では、車両がトラックから外れると罰を受け、ゴールに到達すると報酬が与えられます。時間やスピードが問題ではない場合、これは適切な報酬関数になり得ます。

直線のトラックから外れることなく、できるだけ速く車両を走行させたいと思っています。車両の加速や減速に合わせて、車両は障害物を避け、トラック内に留まっているように、左右にステアリングを操作します。高速でのターンが大きすぎると、簡単に車両がトラックから外れてしまいます。ターンが小さすぎる障害物や別の車両との衝突を回避できない可能性があります。一般的に、最適なアクションは、大きなターンは低速で曲がり、きついカーブではステアリングを少なくすることです。この動作を奨励するため、報酬関数では、高速での小さいターンには正のスコアを割り当てて報酬を与え、高速での大きいターンには負のスコアを割り当てて罰を与える必要があります。同様に、報酬関数では、直線コースでの加速や障害物の近くでの減速に正の報酬を返すことができます。

報酬関数は、AWS DeepRacer モデルの重要な部分です。AWS DeepRacer モデルをトレーニングするときは報酬関数を指定する必要があります。トレーニングでは、最初から最後までトラックでのエピソードが繰り返されます。あるエピソードでは、予想される累積報酬を最大化することで最適な一連のアクションを学習するためにエージェントがトラックを操作します。最後に、トレーニングは強化学習モデルを生成します。トレーニングの後、どのような状態でも最適なアクションができるように、エージェントがモデルで推論を行い、自動運転を実行します。これは、仮想エージェントによるシミュレートされた環境、または AWS DeepRacer スケール車両などの物理的エージェントによる実際の環境下で実行できます。

実際には、強化学習モデルをトレーニングするには、学習アルゴリズムを選択する必要があります。現在、AWS DeepRacer コンソールは、近位ポリシー最適化 ([PPO](#)) と Soft Actor Critic (SAC) アルゴ

リズムのみをサポートしています。その後、最初から記述する場合を除き、選択したアルゴリズムをサポートするディープラーニングフレームワークを選択できます。AWS DeepRacer は SageMaker AI と統合され、[TensorFlow](#) などの一般的な深層学習フレームワークを AWS DeepRacer コンソールですぐに使用できます。フレームワークを使用すると、トレーニングジョブの設定および実行を簡素化し、ユーザーの問題に固有の報酬関数の作成および強化に集中することができます。

トレーニング強化学習モデルは反復プロセスです。まず、ある環境でのエージェントのすべての重要な動作をカバーする報酬関数を一度に定義するのは 困難です。次に、ハイパーパラメータは、多くの場合、十分なトレーニングパフォーマンスを実現するために調整されます。どちらも実験が必要です。賢明なアプローチは、単純な報酬機能から始めて、徐々に強化していくことです。AWS DeepRacer は、トレーニングされたモデルをクローンし、それを駆使して次の のトレーニングで活性化させることで、この反復プロセスを容易にします。各反復で報酬関数に 1 つまたは少数のより洗練された処理を導入して、前に無視された変数を処理するか、結果がコンバートするまでハイパーパラメータを体系的に調整することができます。

機械学習で通常行われるのと同じように、物理エージェントにデプロイして実際の状況での推論を実行する前に、トレーニング済みの強化学習モデルを評価してその有効性を調べる必要があります。自動運転の場合、評価は、車両が最初から最後まで特定のトラックに留まっている頻度やトラックから外れずにコースを完了する速さに基づくことができます。AWS DeepRacer シミュレーションは、評価を実行して、[リーダーボード](#)上の他の AWS DeepRacer ユーザーによってトレーニングされたモデルと比較するため、パフォーマンスのメトリクスを投稿できるようにします。

トピック

- [AWS DeepRacer によるレースの種類とセンサーの有効化](#)
- [AWS DeepRacer コンソールを使用した AWS DeepRacer モデルのトレーニングと評価](#)
- [AWS DeepRacer 報酬関数のリファレンス](#)

AWS DeepRacer によるレースの種類とセンサーの有効化

AWS DeepRacer Leagueでは、以下のタイプのレーシングイベントに参加することができます。

- タイムトライアル: 遮るもののないトラックでタイムを競い合い、可能な限り速いラップタイムを目指します。
- オブジェクト回避: 静止した障害物のあるトラックでタイムを競い合い、可能な限り速いラップタイムを目指します。
- Head-to-bot レース: 同じトラック上の 1 台または複数の他の車両とレースを行い、他の車両よりも前にフィニッシュラインを越えることを目指します。

AWS DeepRacer のコミュニティレースでは現在、タイムトライアルのみをサポートしています。

AWS DeepRacer の車両に搭載する異なったセンサーを実験し、与えられたレースの種類に応じて、周囲を観戦するのに十分な能力を持たせる必要があります。次のセクションでは、[AWS DeepRacer がサポートするセンサー](#)について説明します。これにより、サポートされているタイプの自律型レースイベントを有効化できます。

トピック

- [AWS DeepRacer レーシングタイプのセンサーを選択](#)
- [AWS DeepRacer モデルのトレーニング用にエージェントを設定](#)
- [タイムトライアル用に AWS DeepRacer トレーニングのカスタマイズする](#)
- [オブジェクト回避レース向けの AWS DeepRacer トレーニングをカスタマイズする](#)
- [head-to-bot レース向けの AWS DeepRacer トレーニングをカスタマイズします。](#)

AWS DeepRacer レーシングタイプのセンサーを選択

お客様の AWS DeepRacer の車両には、デフォルトのセンサーとして単眼の前面カメラが付属しています。単眼の前面カメラをもう 1 つ追加して、前面立体視カメラを作成したり、単眼カメラまたは立体視カメラに LiDAR ユニットの追加できます。

以下のリストでは、AWS DeepRacer でサポートされているセンサーの機能と、簡単な費用対効果の分析をまとめています。

前面カメラ

一眼レンズ前面カメラは、トラックボーダーや形状など、ホスト車両前面の環境のイメージをキャプチャできます。最も安価なセンサーで、よくマークされたトラックでの障害物のないタイムトライアルなど、より簡単な自律運転タスクを処理するのに適しています。適切なトレーニングにより、トラック上の固定位置に配置された静止障害物を避けることができます。ただし、障害物の位置情報はトレーニングされたモデルに組み込まれ、その結果、そのモデルが過剰適合され、他の障害物の配置に一般化されない可能性があります。トラック上のランダムな位置に配置された静止オブジェクトまたは他の移動車両では、モデルが収束する可能性は低くなります。

実世界では、AWS DeepRacer の車両にはデフォルトのセンサーとして単眼の前面カメラが搭載されています。カメラには 120 度の広角レンズがあり、RGB イメージをキャプチャし、15 フレーム/秒 (fps) で 160 × 120 ピクセルのグレースケールイメージに変換します。これらのセンサーのプロパティはシミュレーターに保存され、訓練されたモデルがシミュレーションから現実の世界にうまく移行する可能性を最大限に高めます。

前面立体視カメラ

立体視カメラには、同じ解像度と周波数でイメージをキャプチャする複数のレンズがあります。両方のレンズからのイメージは、観察対象のオブジェクトの深度を決定するために使用されます。立体視カメラからの深度情報は、特に動的な環境下で、障害物や前面の他の車両に衝突することを避けるために、ホスト車両にとって役に立ちます。ただし、深度情報が追加されると、トレーニングの収束が遅くなります。

AWS DeepRacer の実車両では、単眼カメラをもう一台追加し、それぞれのカメラを車両の左右に搭載することで、双眼のステレオカメラが構築されています。AWS DeepRacer ソフトウェアは、両方のカメラからの画像キャプチャを同期させます。キャプチャされたイメージは、グレースケールに変換され、積み重ねられ、推論のためにニューラルネットワークに供給されます。シミュレーターでも同じメカニズムが複製され、モデルを実際の環境に一般化するようにトレーニングされます。

LiDAR センサー

LiDAR センサーは、回転レーザーを使用して、可視スペクトル外の光パルスを送信し、各パルスが戻るまでの時間を測定します。特定のパルスが当たるオブジェクトの方向と距離は、LiDAR ユニットを中心とした大きな 3D マップのポイントとして記録されます。

たとえば、LiDAR はホスト車両の死角を検出し、車両の車線変更時の衝突を回避します。LiDAR を単眼カメラまた立体視カメラと組み合わせることで、ホスト車両が適切なアクションを実行するのに十分な情報をキャプチャできるようになります。しかし、LiDAR センサーはカメラに比べてコストが高くなります。ニューラルネットワークは、LiDAR データの解釈方法を学習する必要があります。したがって、トレーニングは収束に時間がかかります。

AWS DeepRacer の実車両では、LiDAR センサーが後部に取り付けられ、6 度傾いています。これは、毎秒 10 回転の角速度で回転し、15cm から 2m の範囲です。ホスト車両の後ろや横にあるオブジェクトだけでなく、前面の車両部品によって遮られていない背の高い物体を検出できます。LiDAR 単位が環境ノイズの影響を受けにくくなるように、角度と範囲が選択されています。

AWS DeepRacer の車両には、対応するセンサーを以下の組み合わせで構成することができます。

- 前面単眼レフカメラのみ。

この設定は、タイムトライアルや固定位置にあるオブジェクトによる障害物の回避に適しています。

- 前面立体視カメラのみ。

この設定は、固定またはランダムな位置にあるオブジェクトとの障害物回避に適しています。

- LiDAR 付き 前面一眼カメラ。

この設定は、障害物回避や head-to-bot レースに適しています。

- LiDAR 付き 前面立体視カメラ。

この設定は、障害物回避や head-to-bot レースには適していますが、タイムトライアルではおそらく最も経済的ではありません。

AWS DeepRacer 車両がタイムトライアルからオブジェクト回避、head-to-bot レースに移行するようにするセンサーを追加することにより車両は環境に関するより多くのデータを収集し、トレーニングの基礎となるニューラルネットワークに供給します。これにより、モデルは複雑さの増加に対処する必要があるため、トレーニングがより困難になります。結局、モデルのトレーニングを学習するタスクはより厳しくなります。

徐々に学習するには、まずタイムトライアルのトレーニングを開始してから、オブジェクト回避に進み、次に head-to-bot レースに進む必要があります。推奨事項の詳細については、次のセクションを参照してください。

AWS DeepRacer モデルのトレーニング用にエージェントを設定

AWS DeepRacer 車両が障害物回避や head-to-bot レースを行うための強化学習モデルをトレーニングするために、適切なセンサーでエージェントを構成する必要があります。簡単なタイムトライアルでは、一眼カメラで設定されたデフォルトのエージェントを使用できます。エージェントの設定では、アクションスペースをカスタマイズし、ニューラルネットワークポロジを選択することで、意図した運転要件を満たすために、選択したセンサーとの連携が向上します。さらに、トレーニング中に視覚的な識別のためにエージェントの外観を変更することもできます。

設定後、エージェント設定は、トレーニングと評価のためにモデルのメタデータの一部として記録されます。評価のために、エージェントは、指定されたセンサー、アクションスペース、ニューラルネットワーク技術を使用するように、記録された設定を自動的に取得します。

このセクションでは、AWS DeepRacer コンソールの中にエージェントを設定するステップについて説明します。

AWS DeepRacer コンソールで AWS DeepRacer エージェントを設定する方法

1. [AWS DeepRacer コンソール](#)にサインインします。

2. プライマリナビゲーションペインで、[ガレージ] を選択します。
3. [ガレージ] を初めて使用する場合は、[ガレージによろこそ] ダイアログボックスが表示されます。AWS DeepRacer 車両でサポートされている各種センサーの概要を参照して [>] または [<] を選択するか [X] を選択してダイアログボックスを閉じます。この入門情報は、[ガレージ] のヘルプパネルで確認できます。
4. [ガレージ] ページで、[新しい車両を構築] を選択します。
5. [自分の車両の修正] ページの [仕様の修正] で、1 つ以上のセンサーを選択して、目的のレーシングタイプに合った最適な組み合わせを試してみてください。

AWS DeepRacer 車両のタイムトライアルをトレーニングするには、[カメラ] を選択します。障害物回避や head-to-bot レースでは、他のタイプのセンサーを使用します。Stereo カメラを選択するために、追加の単眼カメラを取得していることを確認します。AWS DeepRacer は、立体視カメラを 2 つの単眼レフカメラで作成しています。1 台の車両で、単眼カメラまたは二眼ステレオカメラを使用できます。どちらの場合でも、トレーニングされたモデルで障害物回避または head-to-bot レースで死角を検出して回避できるようにする場合は、LiDAR センサーをエージェントに追加できます。

6. [ガレージ] ページの、[ニューラルネットワークトポロジ] で、サポートされているネットワークトポロジを選択します。

一般に、より深いニューラルネットワーク (より多くのレイヤーを持つ) は、急なカーブと多数の曲がり角があるより複雑なトラックを運転や、静止障害物を避けるためにレース、他の移動車両との競争に適しています。しかし、より深いニューラルネットワークはトレーニングにコストがかかり、モデルの収束に時間がかかります。一方、より浅いネットワーク (レイヤーが少ない) では、コストが削減され、トレーニングに要する時間が短縮されます。トレーニングされたモデルは、競争相手がなくても障害のないトラックでのタイムトライアルなど、より単純なトラック条件や運転要件を処理することができます。

具体的には、AWS DeepRacer は、[3-layer CNN] または [5-layer CNN] をサポートしています。

7. [ガレージ] ページで、[次へ] を選択して、エージェントのアクションスペースの設定に進みます。
8. [アクションスペース] ページで、最初のトレーニングのデフォルト設定のままにします。その後のトレーニングでは、ステアリング角度、最高速度、およびそれらの詳細度についてさまざまな設定を試してください。次に、[次へ] を選択します。
9. [群集で目立つように車両を色付け] ページで、[DeepRacer に名前を付ける] に名前を入力し、[車両の色] リストからエージェントの色を選択します。その後で、[送信] を選択します。

10. [ガレージ] ページで、新しいエージェントの設定を確認します。さらに変更を加えるには、[車両の修正] を選択し、[ステップ 4] から前の手順を繰り返します。

これで、エージェントはトレーニングの準備ができました。

タイムトライアル用に AWS DeepRacer トレーニングのカスタマイズする

初めて AWS DeepRacer を使う場合は、まず簡単なタイムトライアルからスタートし、AWS DeepRacer のモデルを学習させて車両を運転する方法に慣れておきます。このようにして、報酬関数、エージェント、環境などの基本的な概念をゆっくり理解することができます。目標は、モデルをトレーニングして車両がトラック上に留まり、できるだけ早くラップを完了するようにすることです。その後、トレーニングしたモデルを AWS DeepRacer 車両にデプロイして、追加のセンサーなしで物理トラックでの走行をテストすることができます。

このシナリオのためにモデルをトレーニングするには、AWS DeepRacer コンソール上の Garage の中からデフォルトエージェントを選択できます。デフォルトのエージェントは、単一の前面カメラ、デフォルトのアクションスペース、およびデフォルトのニューラルネットワークポロジで設定されています。より高度なレースに移行する前に、AWS DeepRacer のデフォルトのエージェントを使用してモデルのトレーニングを開始すると有効です。

デフォルトのエージェントを使用してモデルをトレーニングするには、以下の推奨事項に従ってください。

1. より規則的な形状を持ち、急な曲がり角の少ない単純なトラックで、モデルのトレーニングを開始します。デフォルトの報酬関数を使用します。モデルを 30 分間訓練してください。トレーニングジョブが完了したら、同じトラックでモデルを評価し、エージェントがラップを終えることができるかどうかを確認します。
2. [報酬関数のパラメータ](#) についてお読みください。さまざまなインセンティブを使用してトレーニングを継続し、エージェントに報酬を与えてより速く進みます。次のモデルのトレーニング時間を 1〜2 時間に延長します。最初のトレーニングと 2 番目のトレーニングの報酬グラフを比較します。報酬グラフが改善されなくなるまで実験を続けてください。
3. [アクションスペース](#) の詳細については、こちらをご覧ください。最高速度 (例: 1 m/s) を上げて、3 番目のモデルをトレーニングします。アクションスペースを変更するには、変更を行う機会を得たときに、新しいエージェントを [ガレージ] に構築する必要があります。エージェントの最高速度を更新する場合、最高速度が高いほど、エージェントが評価中のトラックを早く完了でき、AWS DeepRacer 車両が物理トラックでのラップを早く完了できることに注意してください。ただし、最高速度を高くすると、エージェントがカーブ上でオーバーシュートしてトラッ

クから外れてしまう可能性が高くなるため、トレーニングが収束するまでの時間が長くなることがよくあります。精度を低くして、エージェントが加速または減速するための余地を増やし、他の方法で報酬関数を微調整して、トレーニングをより速く収束させることもできます。トレーニングが収束した後、3 番目のモデルを評価して、ラップタイムが改善されるかどうかを確認します。改善されなくなるまで、探索を続けてください。

- より複雑なトラックを選択し、[ステップ 1] から [ステップ 3] を繰り返します。トレーニングに使用したトラックとは異なるトラックでモデルを評価して、モデルが 実際の環境に一般化する さまざまな仮想トラックに一般化できる方法を確認します。
- (オプション) ハイパーパラメータ のさまざまな値を試してトレーニングプロセスを改善し、[ステップ 1] から [ステップ 3] を繰り返します。
- (オプション) AWS DeepRacer ログを調べて分析します。ログの分析に使用できるサンプルコードについては、「<https://github.com/aws-samples/aws-deepracer-workshops/tree/master/log-analysis>」を参照してください。

オブジェクト回避レース向けの AWS DeepRacer トレーニングをカスタマイズする

タイムトライアルに慣れ、いくつかのコンバージドモデルのトレーニングを終えたら、次に厳しい課題である障害物回避に進んでください。ここでは、トラック上に配置されたオブジェクトへのクラッシュを回避しながら、トラックから外れることなく、できるだけ早くラップを完了できるモデルをトレーニングすることを目標としています。これは明らかにエージェントが学習するには難しい問題であり、トレーニングは収束に時間がかかります。

AWS DeepRacer コンソールは、2 種類の障害物回避トレーニングをサポートしています。障害物は、トラック上の固定位置またはランダムな位置に配置できます。固定位置では、障害物はトレーニングジョブ全体を通して同じ場所に固定されたままです。ランダムな場所では、障害物はエピソードごとにそれぞれの場所をランダムに変更されます。

システムの自由度が低いため、位置固定障害物を回避するためにトレーニングが収束しやすくなります。ただし、トレーニングを受けたモデルに位置情報が組み込まれていると、モデルが過剰適合する可能性があります。その結果、モデルが過剰適合し、一般化されないことがあります。ランダムに配置された障害物回避の場合、トレーニングが収束するのが難しくなります。これは、エージェントはこれまでに見たことのない場所で障害物に衝突しないように学習し続ける必要があるためです。ただし、このオプションでトレーニングされたモデルは、より一般化され、実際のレースにうまく移行する傾向があります。まず、障害物を固定位置に配置し、動作に慣れてから、ランダムな位置に取り組みます。

AWS DeepRacer シミュレーターでは、障害物は AWS DeepRacer 車両のパッケージボックスと同じ寸法 (9.5 インチ (長さ) x 15.25 インチ (幅) x 10/5 インチ (高さ)) の直方体ボックスです。これにより、パッケージボックスを物理トラックに障害物として配置すると、訓練されたモデルをシミュレーターから現実の世界へ簡単に移行できます。

障害物回避を試すには、以下のステップで概説されている推奨プラクティスに従ってください。

1. デフォルトのエージェントを使用するか、既存のエージェントをカスタマイズするか、新しいセンサーとアクションスペースを試してみてください。最高速度は 0.8 m/s 以下に、速度の詳細度は 1 または 2 レベルに制限する必要があります。

2 つのオブジェクトを固定したで、約 3 時間のモデルのトレーニングを開始します。報酬機能の例を使用して、レースするトラック、またはそのトラックに非常によく似たトラックでモデルをトレーニングします。「AWS DeepRacer Smile Speedway (中級)」トラックはシンプルなトラックで、サミットレースの準備に適しています。同じ数の障害物がある同じトラックでモデルを評価します。合計予想報酬がどのように収束するかを見てください。

2. [報酬関数のパラメータ](#)についてお読みください。報酬関数のバリエーションを試してみてください。障害物の数を 4 つに増やします。エージェントをトレーニングして、トレーニングが同じトレーニング時間内に収束するかどうかを調べます。収束しない場合は、報酬関数をもう一度微調整し、最高速度を下げるか、障害物の数を減らして、エージェントを再びトレーニングします。大幅な改善がなくなるまで実験を繰り返します。
3. 今度は、ランダムな場所で障害物を回避するトレーニングに進みます。エージェントには、AWS DeepRacer コンソール中のガレージから利用できる追加のセンサーを設定する必要があります。立体視カメラを使用できます。または、LiDAR ユニットの単眼カメラまたは立体視カメラと組み合わせることもできますが、トレーニング時間が長くなります。トレーニングをより速く収束するために、アクションスペースを、比較的低いトップスピード (例: 2 m/s) に設定します。ネットワークアーキテクチャーでは、障害物の回避に十分であることが判明した浅いニューラルネットワークを使用します。
4. 障害物回避のための新しいエージェントで、単純なトラック上にランダムに配置された 4 つのオブジェクトを使用して 4 時間のトレーニングを開始します。次に、同じトラック上でモデルを評価して、ランダムに配置された障害物でラップを終了できるかどうかを確認します。できない場合は、報酬関数を調整し、別のセンサーを試して、トレーニング時間を長くすることができます。別のヒントとして、既存のモデルのクローン作成を試して、以前に学習した経験を活用するためのトレーニングを継続することができます。
5. (オプション) アクションスペースの最高速度を選択するか、トラックに沿ってランダムに配置する障害物を増やします。さまざまなセンサーの組み合わせを試し、報酬関数とハイパーパラメータ値を微調整します。[5 レイヤーの CNN] ネットワークトポロジを試してみてください。

次に、モデルを再トレーニングして、トレーニングの収束にどのような影響を与えるかを判断します。

head-to-bot レース向けの AWS DeepRacer トレーニングをカスタマイズします。

障害物回避のトレーニングを終えたので、次のレベルの課題に取り組む準備ができました。head-to-bot レースのトレーニングモデルです。障害物回避イベントとは異なり、head-to-bot レースは動く車両との動的な環境を持っています。あなたの目標は AWS DeepRacer の車両モデルをトレーニングして、他の移動車両と競争し、トラックから外れたり、他の車両のいずれかと衝突することなく、最初にフィニッシュラインに到達することです。AWS DeepRacer コンソールでは、エージェントに 1-4 ボットの車両と競争させることで、head-to-bot レースモデルのトレーニングすることができます。一般的に言えば、より長いトラックにより多くの障害物を配置する必要があります。

各ボット車両は、一定の速度で事前に定義された経路をたどります。車線を変更したり、開始車線に残ったりすることができます。障害物回避のトレーニングと同様に、ボット車両を両車線のトラック全体に均等に分散させることができます。コンソールでは、トラック上のボット車両を、最大 4 台までと制限しています。トラック上により多くの競合車両を置くことにより、学習エージェントは、他の車両との多様な状況に遭遇する機会が増えます。この方法では、1 つのトレーニングジョブでより多くの学習を行われ、エージェントはより速くトレーニングを受けることができます。ただし、各トレーニングは収束に時間がかかる可能性があります。

ボット車両を持つエージェントをトレーニングするには、エージェントのアクションスペースの最高速度をボット車両の (一定の) 速度よりも高く設定して、エージェントがトレーニング中に通過する機会を増やす必要があります。まず、エージェントの最高速度を 0.8 m/s に設定し、ボット車両の移動速度を 0.4 m/s に設定してください。ボットが車線を変更できるようにすると、エージェントは同じ車線の前面の移動車両に衝突しないようにする方法だけでなく、他のレーンの前面の別の移動車両に衝突しないようにする方法も学習する必要があるため、トレーニングはより困難になります。ボットは、ランダムな間隔で車線を変更するように設定できます。インターバルの長さは、トレーニングジョブを開始する前に指定した時間の範囲 (例: 1 秒 ~ 5 秒) からランダムに選択されます。この車線変更の動作は、現実世界の head-to-bot レースの動作に似ており、トレーニングを受けたエージェントはより良く生成するはずですが、モデルのトレーニングの収束には時間がかかります。

次の推奨ステップに従って、head-to-bot レースのトレーニングを繰り返してください。

1. AWS DeepRacer コンソールのガレージに、ステレオカメラと LiDAR 単位の両方で構成された新しいトレーニングエージェントを構築します。ボット車両に対して立体視カメラのみを使用して、比較的優れたモデルをトレーニングすることができます。LiDAR は、エージェントが車線

を変更したときに死角を減らすのに役立ちます。最高速度を高く設定しすぎないでください。最適な開始ポイントは、1 m/s です。

2. Head-to-bot レースをトレーニングするには、2 台のボット車から始めます。ボットの移動速度をエージェントの最高速度よりも低く設定します (例: エージェントの最高速度が 1 m/s の場合、0.5 m/s に設定)。車線変更オプションを無効にし、先ほど作成したトレーニングエージェントを選択します。報酬関数の例の 1 つを使用するか、最小限の修正を行い、3 時間トレーニングします。レースに参加するトラック、またはそのトラックに非常によく似たトラックを使用します。「AWS DeepRacer Smile Speedway (中級)」トラックはシンプルなトラックで、サミットレースの準備に適しています。トレーニングが完了したら、トレーニングされたモデルを同じトラックで評価します。
3. より困難なタスクに備えて、2 番目の Head-to-bot レースモデル向けにトレーニングされたモデルをクローンします。さらに多くのボット車両を試すか、車線変更オプションを有効にしてください。2 秒以上のランダムな間隔で遅い車線変更オペレーションを開始します。また、カスタム報酬関数を試してみたいかもしれません。一般に、カスタムの報酬関数のロジックは、他の車両を上回って軌道に乗っている間のバランスを考慮しない場合、障害物回避のロジックと似ています。以前のモデルがどれほど良いかによって、さらに 3 ~ 6 時間をトレーニングする必要があるかもしれません。モデルを評価し、モデルの実行方法を確認します。

AWS DeepRacer コンソールを使用した AWS DeepRacer モデルのトレーニングと評価

強化学習モデルをトレーニングするには AWS DeepRacer コンソールを使用できます。コンソールで、トレーニングジョブを作成し、サポートされているフレームワークと利用可能なアルゴリズムを選択し、報酬関数を追加して、トレーニング設定を設定します。シミュレーターでトレーニングの進捗状況を確認することもできます。詳細な手順については、「[the section called “最初のモデルのトレーニング”](#)」を参照してください。

このセクションでは、AWS DeepRacer モデルの学習方法と評価方法について説明します。また、報酬関数の作成方法と改善方法、アクションスペースがモデルのパフォーマンスに与える影響、ハイパーパラメータがトレーニングのパフォーマンスに与える影響についても説明します。また、トレーニングモデルのクローンを作成してトレーニングセッションを延長する方法、シミュレーターを使用してトレーニングのパフォーマンスを評価する方法、およびシミュレーションの一部を実世界の課題に対処させる方法についても学習できます。

トピック

- [報酬関数を作成する](#)

- [堅牢なモデルをトレーニングするためのアクションスペースを探す](#)
- [ハイパーパラメータを体系的に調整する](#)
- [AWS DeepRacer Training トレーニングジョブの進行状況を調べる](#)
- [トレーニングモデルを複製して新しいトレーニングパスを開始する](#)
- [シミュレーションでの AWS DeepRacer モデルを評価します](#)
- [実環境に合わせた AWS DeepRacer モデルのトレーニングを最適化する](#)

報酬関数を作成する

報酬関数とは、AWS DeepRacer の車両がトラック上のある位置から新しい位置に移動したときの、即座のフィードバック (報酬またはペナルティスコアとして) を表します。この関数の目的は、事故や違反なく、目的の場所にすばやく到着するために車両をトラックに沿って移動させることです。望ましい動きはそのアクションまたはその目標状態に対してより高い得点を得ることができます。違法な動きまたは無駄な動きをするとより低いスコアになります。AWS DeepRacer モデルをトレーニングする場合、報酬関数は唯一のアプリケーション固有の部分になります。

一般的に、報酬関数はインセンティブプランのように機能するように設計します。インセンティブ戦略が異なると、車両の動作が異なる可能性があります。車両をより速く走らせるために、この関数が車両がトラックに沿って走行することに対して報酬を与える必要があります。この関数は、車両がラップを終了するのに時間がかかりすぎたり、トラックから外れたときにペナルティを課す必要があります。ジグザグな運転パターンを避けるために、トラックのまっすぐな部分ではあまりステアリングを使用しない車両に報酬を与える場合があります。によって測定されるように、車両が特定のマイルストーンを通過すると、報酬関数はプラスのスコアを与える場合があります。[waypoints](#)これは待機や間違った方向への運転を軽減する可能性があります。また、トラックのコンディションを考慮して報酬関数を変更する場合があります。ただし、報酬関数が環境固有の情報を考慮に入れるほど、トレーニングされたモデルが過適合となり、汎用性が失われます。モデルをより一般的に適用可能にするために、[アクションスペース](#)を探索することができます。

インセンティブプランは慎重に考慮されない場合、[意図しない反対の結果](#)につながる可能性があります。これは強化学習にとっては即時のフィードバックが必要ですが、条件が十分ではないために可能性があります。個々の即時報酬自体も、その移動が必要なものであるかどうかを判断できません。指定された位置で、移動は高い報酬を獲得します。その後の移動によってトラックから外れ、低得点になる可能性があります。そのような場合、車両はその位置でのハイスコアの移動を避ける必要があります。任意の位置からの今後のすべての移動が平均して高いスコアを生み出すと見込まれる場合限り、次の位置への移動が望ましいと見なされます。今後のフィードバックは、今後の少数の移動または平均報酬計算に含まれる位置を許容するレートで割り引かれます。

[報酬関数](#)を作成する際は、基本的なシナリオをカバーした簡単なものから始めることをお勧めします。より多くのアクションを処理するように関数を拡張することができます。それでは、いくつかの簡単な報酬関数を見てみましょう。

トピック

- [報酬関数の簡単な例](#)
- [自分の報酬関数を強化する](#)

報酬関数の簡単な例

最も基本的な状況を最初に検討することによって、報酬関数の構築を開始することができます。トラックから外れることなく最初から最後まで直線的なトラックを運転しているという状況です。このシナリオでは、報酬関数ロジックは `on_track` と `progress` のみに依存します。トライアルとして、次のロジックから始めることができます。

```
def reward_function(params):
    if not params["all_wheels_on_track"]:
        reward = -1
    else if params["progress"] == 1 :
        reward = 10
    return reward
```

このロジックでは、エージェントがトラックから外れるとエージェントにペナルティが課されます。エージェントが終了地点に到達すると報酬が与えられます。指定された目標を達成するために合理的です。ただし、エージェントは、トラックを逆方向に進むことも含め、開始地点と終了地点の間を自由に移動します。トレーニングを完了するのに長い時間がかかるだけでなく、トレーニングされたモデルでも、実世界の車両にデプロイした場合に運転効率が低下する可能性があります。

実際には、エージェントは、トレーニングのコースを通して少しずつ学習できれば、より効果的に学習します。これは、報酬関数がトラックに沿って段階的に小さい報酬を与える必要があることを意味します。エージェントが直線的なトラックを走るためには、次のように報酬関数を改善することができます。

```
def reward_function(params):
    if not params["all_wheels_on_track"]:
        reward = -1
    else:
        reward = params["progress"]
```

```
return reward
```

この関数では、エージェントはフィニッシュラインに近づくほど報酬が増えます。これによつては非生産的な逆方向への運転の試行を減らすか排除するはずでありません。一般的に、私たちは報酬関数がアクション空間により均一に報酬を分配することを望みます。効果的な報酬関数を作成することは、困難な仕事です。簡単な関数から始めて、徐々に強化または改善してください。体系的な実験により、関数はより堅牢で効率的になります。

自分の報酬関数を強化する

AWS DeepRacer モデルをシンプルな直線的なトラック用に正常にトレーニングすると、AWS DeepRacer 車両 (仮想または物理的) がトラックから外れることなく走行できるようになります。ループされたトラック車両で走らせると、トラック上にとどまりません。報酬関数はトラックをたどるためにターンをするアクションを無視しました。

車両にこれらのアクションに対応させるためには、報酬関数を強化する必要があります。エージェントが許容されるターンを行った場合にはその関数は報酬を与え、エージェントが違法なターンを行った場合にはペナルティを課す必要があります。別のラウンドのトレーニングを開始する準備が整いました。以前のトレーニングを活用するには、以前にトレーニングされたモデルをクローンして、以前に学習した知識を引き継いで新しいトレーニングを開始できます。このパターンに従って、ますます複雑な環境で運転する AWS DeepRacer 車両をトレーニングするために、報酬関数にさらに機能を追加することができます。

より高度な報酬関数については、以下の例を参照してください。

- [the section called “例 1: タイムトライアルでセンターラインに従う”](#)
- [the section called “例 2: タイムトライアルで 2 つの境界内に留まる”](#)
- [the section called “例 3: タイムトライアルでのジグザグ運転の防止”](#)
- [the section called “例 4: 静止している障害物や走行中の車両に衝突することなく、1 つの車線に留まること。”](#)

堅牢なモデルをトレーニングするためのアクションスペースを探す

原則として、できるだけ多くの環境に適用できるように、モデルをできる限り堅牢になるようにトレーニングしてください。堅牢なモデルは、広範囲のトラックの形状や条件に適用できるものです。一般的に、堅牢なモデルはその報酬関数が明示的な環境特有の知識を含む能力を持っていないので「スマート」ではありません。そうでなければ、モデルはトレーニングされた環境に似た環境にのみ適用可能である可能性があります。

環境固有の情報を報酬関数に明示的に組み込むことは、特徴量エンジニアリングに相当します。特徴量エンジニアリングはトレーニング時間を短縮するのに役立ち、特定の環境に合わせて作られたソリューションに役立ちます。ただし、一般的な適用性のモデルをトレーニングするには、多くの特徴量エンジニアリングを試みることを控える必要があります。

たとえば、循環トラックでモデルをトレーニングする場合、そのような幾何学的プロパティを明示的に報酬関数に組み込んでいる場合は、循環以外のトラックに適用可能なトレーニング済みモデルを入手することは期待できません。

報酬関数を可能な限りシンプルに保ちながら、モデルをできる限り頑健にトレーニングする方法について教えてください。1つの方法はエージェントが取ることができるアクションにまたがるアクションスペースを探すことです。もう1つの方法は、基礎となるトレーニングアルゴリズムの[ハイパーパラメータ](#)で実験することです。多くの場合、両方を使用します。ここでは、アクションスペースを探索して、AWS DeepRacer 車両の堅牢なモデルをトレーニングする方法に焦点を当てます。

AWS DeepRacer モデルのトレーニングでは、アクション (a) は速度 (t メートル/秒) とステアリング角度 (s 度) の組み合わせになります。エージェントのアクションスペースは、エージェントが取り得る速度とステアリングの範囲を定義します。速度の m 数、 (v_1, \dots, v_n) およびステアリングの n 数、 (s_1, \dots, s_m) の個別のアクションスペースの場合、アクションスペースには $m \times n$ の可能なアクションがあります。

```

a1:          (v1, s1)
...
an:          (v1, sn)

...
a(i-1)*n+j: (vi, sj)
...

a(m-1)*n+1: (vm, s1)
...
am*n:       (vm, sn)

```

(v_i, s_j) の実際の値は v_{\max} と $|s_{\max}|$ の範囲によって異なり、均一に分布しているわけではありません。

お客様の AWS DeepRacer モデルのトレーニングを開始するかまたは反復するたびに、まず n 、 m 、 v_{\max} 、および $|s_{\max}|$ を指定するか、それらのデフォルト値を使用することに同意する必要があります。選択に基づいて、AWS DeepRacer サービスはエージェントがトレーニングで選択でき

る利用可能なアクションを生成します。生成されたアクションは、アクションスペース全体に均一に分布しているわけではありません。

一般的に、より多くのアクションとより大きなアクション範囲は、不規則な回転角や方向を持つ曲線トラックなど、より多様なトラック条件に対応するためのより多くのスペースまたはオプションをエージェントに提供します。エージェントが利用できるオプションが多ければ多いほど、トラックのバリエーションをより簡単に処理できます。その結果、単純な報酬関数を使用している場合でも、トレーニング済みモデルがより広く適用可能になることが期待できます。

たとえば、エージェントは、わずかな速度とステアリング角で荒削りなアクションスペースを使用した直線トラックに対応することを素早く学習することができます。曲線トラックでは、この荒削りなアクションスペースのために、エージェントは行き過ぎて、ターンする際にトラックから外れる可能性があります。これは、速度やステアリングを調整するためのオプションが十分でないためです。速度またはステアリングの数、あるいはその両方を増やすことにより、エージェントは、トラックに沿いながらカーブに対応できるようになります。同様に、エージェントがジグザグに動く場合は、任意のステップでステアリング範囲の数を増やすことを試みることで急激なターンを減らすことができます。

アクションスペースが大きすぎると、アクションスペースを探索するのにより長い時間がかかるため、トレーニングのパフォーマンスが低下する可能性があります。モデルの一般的な適用性のメリットとトレーニングのパフォーマンス要件とのバランスをとるようにしてください。この最適化には体系的な実験が含まれています。

ハイパーパラメータを体系的に調整する

モデルのパフォーマンスを向上させる 1 つの方法は、より優れた、またはより効果的なトレーニングプロセスを実行することです。たとえば、堅牢なモデルを取得するには、トレーニングによって、エージェントのアクションスペース全体にわたって、エージェントが多かれ少なかれ均等に分散したサンプリングを提供する必要があります。これには、探査と搾取の十分な組み合わせが必要です。これに影響を与える変数には、使用されるトレーニングデータの量 (number of episodes between each training および batch size)、エージェントがどれだけ早く学習できるか (learning rate)、探索の一部 (entropy) が含まれます。実用的なトレーニングを行うには、学習プロセスを高速化する必要があります。これに影響を与える変数には、learning rate、batch size、number of epochs、discount factor があります。

トレーニングプロセスに影響を与える変数は、トレーニングのハイパーパラメータとして知られています。これらのアルゴリズムの属性は、基盤となるモデルのプロパティではありません。残念ながら、ハイパーパラメータは本質的に経験的なものです。これらの最適値は、すべての実用的な目的に対して知られているわけではなく、導き出すために体系的な実験を必要とします。

AWS DeepRacer モデルのトレーニングのパフォーマンスを調整するために調整できるハイパーパラメータについて説明する前に、次の用語を定義しましょう。

データポイント

データポイントは、経験としても知られており、 (s,a,r,s') の連符です。ここで、 s はカメラによってキャプチャされた模様 (または状態)、 a は車両によって実行されたアクションを表します。 r は、その行動によってもたらされる予想される報酬のためのものであり、そして s' はその行動がとられた後の新しい模様のためのものです。

エピソード

エピソードとは、車両が任意の出発点から出発し、最終的にトラックを完走するかまたはトラックから外れるまでの期間です。これにより一連のエクスペリエンスが具体化されます。エピソードごとに長さが異なる場合があります。

エクスペリエンスバッファ

エクスペリエンスバッファは、トレーニング中にさまざまな長さの一定数のエピソードにわたって収集された多数の順序付けられたデータ点から構成されています。AWS DeepRacer の場合、それは AWS DeepRacer 車両に搭載されたカメラによって捉えられた画像と車両によって取られたアクションに対応し、基盤となる (ポリシーと値) ニューラルネットワークを更新するための入力が引き出される情報源として機能します。

バッチ

バッチは、ポリシーネットワークの重みを更新するために使用される、一定期間にわたるシミュレーションの一部を表すエクスペリエンスの順序付きリストです。これはエクスペリエンスバッファのサブセットです。

トレーニングデータ

トレーニングデータは、エクスペリエンスバッファからランダムにサンプリングされ、ポリシーネットワークの重みをトレーニングするために使用されるバッチのセットです。

アルゴリズムハイパーパラメータとその効果

ハイパーパラメータ	説明
勾配降下のバッチサイズ	<p>最近の車両エクスペリエンスの数はエクスペリエンスバッファから無作為に抽出され、基盤となる深層学習ニューラルネットワークの重みを更新するために使用されます。無作為抽出は、入力データに内在する相関関係を低減するのに役立ちます。より大きなバッチサイズを使用して、ニューラルネットワークの重みをより安定してスムーズに更新するようにしますが、トレーニングが長くなったり遅くなったりする可能性があるので注意してください。</p> <p>必須</p> <p>はい</p> <p>有効値</p> <p>(32, 64, 128, 256, 512) の正の整数</p> <p>デフォルトの値</p> <p>64</p>
[エポック数]	<p>勾配降下中にニューラルネットワークの重みを更新するためにトレーニングデータを通る回数。トレーニングデータはエクスペリエンスバッファからの無作為抽出に対応します。より安定した更新を促進するために多数のエポックを使用しますが、トレーニングが遅くなることが予想されます。バッチサイズが小さい場合は、少数のエポックを使用できます。</p> <p>必須</p> <p>いいえ</p> <p>有効値</p> <p>[3 - 10] 間の正の整数</p> <p>デフォルトの値</p> <p>3</p>
[学習レート]	<p>各更新中に、新しい重みの一部は勾配降下 (または上昇) の寄与から得られ、残りは既存の重みの値から得られます。学習レートは、勾配降下 (または上昇) の更新</p>

ハイパーパラメータ	<p data-bbox="342 142 407 176">説明</p> <p data-bbox="342 258 1494 432">がネットワークの重みにどれだけ寄与するかを制御します。より高い学習レートを使用してより速いトレーニングのための勾配降下寄与をより多く含めますが、学習レートが大きすぎると予想される報酬が収束しない可能性があることに注意してください。</p> <p data-bbox="342 480 407 514">必須</p> <p data-bbox="391 562 485 596">いいえ</p> <p data-bbox="342 617 440 651">有効値</p> <p data-bbox="391 693 1317 730">0.00000001 (または 10^{-8}) と 0.001 (または 10^{-3}) の間の実数</p> <p data-bbox="342 751 566 785">デフォルトの値</p> <p data-bbox="391 833 505 867">0.0003</p>
Entropy	<p data-bbox="342 913 1494 1087">ポリシー配布に無作為性を追加するタイミングを決定するために使用されるある程度の不確実性。不確実性が増したことで、AWS DeepRacer 車両はアクションスペースをより広く探索することができます。エントロピー値が大きいほど、車両はアクションスペースをより徹底的に探索します。</p> <p data-bbox="342 1136 407 1169">必須</p> <p data-bbox="391 1218 485 1251">いいえ</p> <p data-bbox="342 1272 440 1306">有効値</p> <p data-bbox="391 1354 662 1388">0 と 1 の間の実数。</p> <p data-bbox="342 1409 566 1442">デフォルトの値</p> <p data-bbox="391 1491 469 1524">0.01</p>

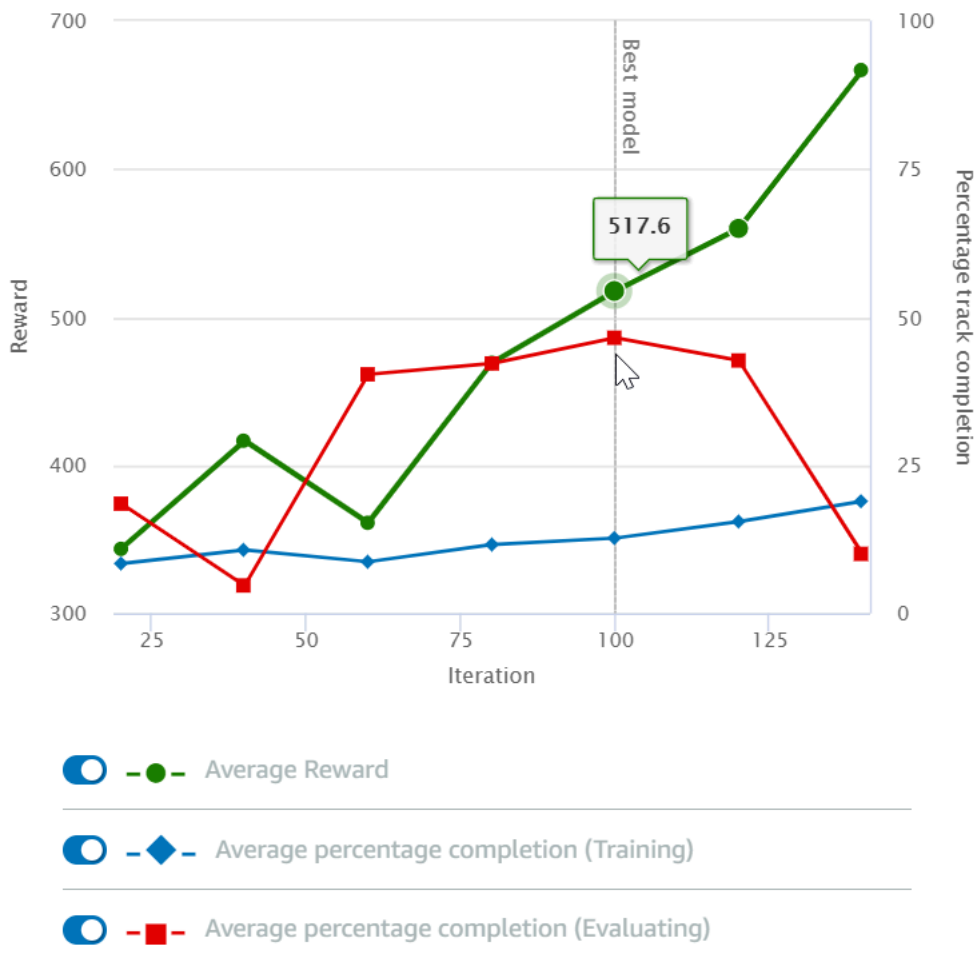
ハイパーパラメータ	説明
[割引係数]	<p>係数は、将来の報酬が期待される報酬にどのくらい寄与するかを指定します。割引係数の値が大きいほど、車両がアクションを実行するとみなしている寄与の範囲が広くなり、トレーニングが遅くなります。割引係数が 0.9 の場合、移動のために将来の 10 ステップのオーダーからの報酬が含まれます。0.999 の割引係数で、車両は移動をするために将来の 1000 ステップのオーダーからの報酬を考慮します。推奨される割引係数値は、0.99、0.999 および 0.9999 です。</p> <p>必須</p> <p>いいえ</p> <p>有効値</p> <p>0 と 1 の間の実数。</p> <p>デフォルトの値</p> <p>0.999</p>

ハイパーパラメータ	説明
[損失タイプ]	<p>ネットワークの重みを更新するために使用される目標関数のタイプ。優れたトレーニングアルゴリズムは、エージェントの戦略を徐々に変化させて、ランダムなアクションをとることから、報酬を増やすための戦略的なアクションをとることへと徐々に移行するべきです。しかし、それがあまりにも大きな変化をするならば、トレーニングは不安定になりエージェントは学習しません。[Huber 損失]と[平均二乗誤差]のタイプは、小規模な更新でも同様に動作します。しかし、アップデートが大きくなるにつれ、[Huber 損失]と[平均二乗誤差損失]に比べて増分が小さくなります。収束の問題がある場合は、[Huber 損失]タイプを使用してください。収束性が良く、より早くトレーニングする場合は、[平均二乗誤差]タイプを使用してください。</p> <p>必須</p> <p>いいえ</p> <p>有効値</p> <p>([Huber 損失]、[平均二乗誤差])</p> <p>デフォルトの値</p> <p>[Huber 損失]</p>

ハイパーパラメータ	説明
[各ポリシー更新反復間のエクスペリエンスエピソードの数]	<p>ポリシーネットワークの重み付けを学習するためのトレーニングデータを取得するために使用されるエクスペリエンスバッファのサイズ。エピソードとは、エージェントが任意の出発点から出発し、最終的にトラックを完走するかまたはトラックから外れるまでの期間です。一連のエクスペリエンスで構成されます。エピソードごとに長さが異なる場合があります。単純な強化学習問題では、エクスペリエンスバッファが少なくても済み、学習は速いです。より多くのローカル最大値のあるより複雑な問題では、より多くの相関性のないデータポイントを提供するためにより大きなエクスペリエンスバッファが必要です。この場合、トレーニングは遅くなりますが、安定しています。推奨される値は 10、20 および 40 です。</p> <p>必須</p> <p>いいえ</p> <p>有効値</p> <p>5 と 100 の間の整数</p> <p>デフォルトの値</p> <p>20</p>

AWS DeepRacer Training トレーニングジョブの進行状況を調べる

トレーニングジョブを開始した後、エピソードごとの報酬とトラック完走状況のトレーニングメトリクスを調べ、モデルのトレーニングジョブのパフォーマンスを確認できます。AWS DeepRacer コンソールでは、メトリクスは以下の図に示すように「報酬グラフ」に表示されます。

Reward graph [Info](#)

エピソードごとに獲得した報酬、反復ごとの平均報酬、エピソードごとの進行状況、反復ごとの平均進行状況、またはそれらの任意の組み合わせを表示することを選択できます。これを行うには、[報酬 (エピソード、平均)] または [報酬グラフ] の最下部にある [進行状況 (エピソード、平均)] スイッチを切り替えます。エピソードごとの報酬と進行状況は、異なる色の散布図として表示されます。平均の報酬とトラック完走状況は、ラインプロットで表示され、最初の反復後に開始されます。

報酬の範囲はグラフの左側に表示され、進行状況の範囲 (0 ~ 100) は右側に表示されます。トレーニングメトリクスの正確な値を読み取るには、グラフ上のデータポイントの近くまでマウスを移動します。

トレーニングの進行中、グラフは 10 秒ごとに自動的に更新されます。更新ボタンを選択して、メトリクス表示を手動で更新できます。

トレーニングジョブは、平均の報酬とトラック完走状況が収束する傾向を示している場合に適しています。特に、エピソードごとの進行状況が継続的に 100% に達し、報酬レベルが横ばいになる場合、モデルが収束している可能性があります。それ以外の場合は、モデルを複製して再トレーニングします。

トレーニングモデルを複製して新しいトレーニングパスを開始する

トレーニングの新しいラウンドの開始点として、以前にトレーニングしたモデルを複製すると、トレーニングの効率を向上させることができます。これを行うには、すでに学んだ知識を利用するようにハイパーパラメータを変更します。

このセクションでは、AWS DeepRacer コンソールを使用してトレーニング済みモデルを複製する方法を学びます。

AWS DeepRacer コンソールを使用して強化学習モデルのトレーニングを繰り返すには

1. まだサインインしていない場合は、AWS DeepRacer コンソールにサインインします。
2. [モデル] ページで、トレーニングされたモデルを選択し、次に [アクション] ドロップダウンメニューリストから [クローン] を選択します。
3. [モデルの詳細] で、以下の操作を行います。
 - a. クローンモデルに対して名前を生成したくない場合は、[モデル名] に `RL_model_1` をタイプします。
 - b. 必要に応じて、[モデルの説明 - オプション] に複製したモデルの説明を入力します。
4. 環境シミュレーションのために、別のトラックのオプションを選択します。
5. [報酬関数] で、利用可能な報酬関数の例の 1 つを選択してください。報酬関数を変更します。たとえば、ステアリングを検討してください。
6. [アルゴリズム設定] を展開し、さまざまなオプションを試してください。たとえば、[勾配降下バッチサイズ] の値を 32 から 64 に変更するか、[学習レート] を上げてトレーニングをスピードアップします。
7. [条件の停止] のさまざまな選択を試してください。
8. [トレーニングの開始] を選択して、新しいトレーニングを開始します。

一般的な堅牢な機械学習モデルのトレーニングと同様に、最良の解決策を見つけるために体系的な実験を行うことが重要です。

シミュレーションでの AWS DeepRacer モデルを評価します

モデルを評価するには、トレーニング済みのモデルのパフォーマンスをテストします。AWS DeepRacer では、標準的なパフォーマンスメトリクスは、3 回の連続ラップを終えた平均時間になります。任意の 2 つのモデルについて、このメトリクスを使用すると、エージェントが同じトラックで他のモデルより速く走行できる場合、一方のモデルが優れています。

一般的に、モデルの評価には次の作業が含まれます。

1. 評価ジョブを設定し、起動します。
2. ジョブの実行中に進行中の評価を確認してください。AWS DeepRacer シミュレーターでこれを行うことができます。
3. 評価ジョブが完了したら、評価の概要を確認してください。進行中の評価ジョブはいつでも終了できます。

Note

評価時間は、選択した基準によって異なります。モデルが評価基準を満たさない場合、評価は 20 分の上限に達するまで継続されます。

4. 必要に応じて、評価結果を適格な [AWS DeepRacer リーダーボード](#) に送信してください。リーダーボードでのランキングによって、自分のモデルが他の参加者に対してどの程度うまく機能しているかがわかります。

物理的なトラックで走行している AWS DeepRacer 車両で AWS DeepRacer モデルをテストしてください。「[車両を運転する](#)」を参照してください。

実環境に合わせた AWS DeepRacer モデルのトレーニングを最適化する

[アクションスペースの選択](#)、[報酬関数](#)、トレーニングで使用される[ハイパーパラメータ](#)、[車両のキャリブレーション](#)、さらには[実際のトラック](#)状態を含め、トレーニングされたモデルの実世界でのパフォーマンスには多くの要因が影響します。また、シミュレーションは実世界の (しばしば未処理の) 近似にすぎません。課題となるのは、シミュレーションでモデルをトレーニングし、それを実世界に適用し、そして満足のいく性能を達成することです。

実世界での堅実なパフォーマンスを得るためにモデルをトレーニングするには、多くの場合、[報酬関数](#)、[アクションスペース](#)、[ハイパーパラメータ](#)、およびシミュレーションでの[評価](#)および実環境で

の[テスト](#)という反復作業が多数必要になります。最後のステップは、いわゆるシミュレーションから実社会への移行 (sim2real) であり、扱いにくいと感ずることがあります。

sim2real の課題に取り組むためには、次の考慮事項に注意してください。

- 車両が適切にキャリブレートされていることを確認します。

シミュレーション環境は実環境を部分的に表現したものであるため、これは重要です。また、エージェントは、各ステップで、カメラのイメージによってキャプチャされた現在のトラック状態に基づいてアクションを実行します。高速ではルートを計画するのに十分遠くまで見ることはできません。これに対応するために、シミュレーションは速度とステアリングに制限を課します。トレーニングされたモデルが実世界で機能するために、車両はこれと他のシミュレーション設定に合うように適切にキャリブレートされなければなりません。車体のキャリブレーションの詳細については、「[the section called “車両をキャリブレートする”](#)」を参照してください。

- 最初にデフォルトのモデルで車両をテストします。

お客様の AWS DeepRacer 車両には、その推理エンジンにロードされたトレーニング済みモデルが付属しています。実世界で自身のモデルをテストする前に、車両がデフォルトのモデルで合理的にうまく機能することを確認してください。そうでない場合は、物理的なトラックセットアップを確認します。正しく構築されていない物理トラックでモデルをテストすると、パフォーマンスが低下する可能性があります。このような場合は、テストを開始または再開する前にトラックを再設定または修復します。

Note

AWS DeepRacer 車両を走行させる際、報酬機能呼び出すことなくトレーニングされたポリシーネットワークに従ってアクションが推論されます。

- モデルがシミュレーションで機能することを確認します。

モデルが実世界でうまく機能しない場合は、モデルかトラックのどちらかに欠陥がある可能性があります。根本原因を整理するには、まず[シミュレーションでモデルを評価して](#)、シミュレーションされたエージェントがトラックから外れることなく少なくとも 1 回のループを終了できるかどうかを確認する必要があります。シミュレーターでエージェントの軌跡を観察しながら、報酬の集約を調べることでこれを実行できます。シミュレーションされたエージェントが失敗することなくループを完了したときに報酬が最大に達する場合、そのモデルが適切である可能性があります。

- モデルを過剰にトレーニングしないでください。

モデルが一貫してシミュレーションでトラックを完了した後でトレーニングを続けると、モデルに過剰適合が発生します。シミュレーションされたトラックと実環境との間のわずかな違いでさえも処理できないため、過剰にトレーニングされたモデルは実世界ではうまく機能しません。

- さまざまな反復から複数のモデルを使用します。

一般的なトレーニングセッションでは、過小適合と過剰適合の間にあるさまざまなモデルが作成されます。正しいモデルを決定するための先験的な基準がないため、エージェントがシミュレーターで1回のループを完走してからループを一貫して実行されるまでの間に、いくつかのモデル候補を選択する必要があります。

- テストではゆっくりと走り始め、徐々に走行速度を上げます。

車両にデプロイされたモデルをテストする場合は、小さい最大速度値から開始します。たとえば、テストの制限速度をトレーニングの制限速度の10%未満に設定できます。その後、車両が動き始めるまで、テストの制限速度を徐々に上げます。装置制御コンソールを使用して車両をキャリブレーションするときに、テストの制限速度を設定します。車両の速度が速すぎる場合、たとえば、速度がシミュレーターでのトレーニング中に見られる速度を超える場合、モデルは実際のトラックでうまく機能する可能性は低くなります。

- さまざまな開始位置で車両をテストします。

モデルはシミュレーションで特定の経路をたどることを学習し、トラック内の位置に敏感に反応できるようになります。モデルが特定の位置からうまく機能するかどうかを確認するには、トラック境界内のさまざまな位置(左から中央、右)から車両テストを開始する必要があります。ほとんどのモデルは、車両を白線のどちらかの側に近づける傾向があります。車両の経路を分析するために、シミュレーションからステップごとに車両の位置(x、y)をプロットして、実環境で車両がたどる可能性のある経路を特定します。

- 直線トラックでテストを開始します。

直線トラックは、曲線トラックに比べて移動がはるかに簡単です。まっすぐなトラックでテストを開始すると、パフォーマンスの低いモデルをすばやく取り除くことができます。ほとんどの場合、車両が直線トラックに沿って走行できない場合、そのモデルは曲線トラックでもうまく機能しません。

- 車両が1種類のアクションしかとらない行動に注意します。

車両が1種類のアクションのみを実行できる場合、たとえば、車両を左方向にのみハンドルを切る場合、モデルは過剰適合か過小適合の可能性があり、指定されたモデルパラメータでは、ト

レーニングの反復が多すぎるとモデルが過剰適合になる可能性があります。反復回数が少なすぎると、過小適合になる可能性があります。

- トラックの境界線に沿って進路を修正する車両の能力に注意します。

良いモデルは、トラックの境界に近づいたときに車両が自身を修正します。適切にトレーニングされたモデルのほとんどが、この機能を備えています。車両がトラックと境界の両方で適切な動作をする場合は、モデルはより頑強でより高品質であると見なされます。

- 車両が示す矛盾した行動に注意します。

ポリシーモデルは、任意の状態で行動をとる確率分布を表します。トレーニングされたモデルが推論エンジンにロードされると、車両はモデルの処方に従って、最も可能性の高い行動を1ステップずつ選択します。行動確率が均等に分布している場合、車両は等しいまたは非常に類似した確率の行動のいずれかをとる可能性があります。これは不安定な運転行動につながります。たとえば、車両が時々(たとえば、しばしば)直線経路をたどり、他の時には不必要な方向転換をする場合は、モデルは過小適合又は過剰適合のいずれかです。

- 車両によって行われる1種類のターン(左または右)のみに注意してください。

車両が左には非常にうまく曲がるが、右へのステアリングを管理できない場合、または同様に、右にはうまく曲がれるが左にはうまく曲がれない場合は、車両のステアリングをキャリブレートするか、再キャリブレートします。あるいは、テスト中の物理的設定に近い設定でトレーニングされたモデルを使用することもできます。

- 急なターンを行ってトラックから外れる車両に注意します。

車両がほとんどの場合正しい経路をたどるが、突然トラックから外れる場合は、おそらく環境にある集中力をそらすものが原因の可能性があります。集中力をそらすものの多くには、予期しないまたは意図しないライトの反射が含まれます。そのような場合は、トラックのまわりに障壁を使用するかその他の方法でまぶしい光を低減します。

AWS DeepRacer 報酬関数のリファレンス

AWS DeepRacerの報酬関数に関する技術的なリファレンスは以下の通りです。

トピック

- [AWS DeepRacer 報酬関数の入力パラメータ](#)
- [AWS DeepRacer 報酬関数の例](#)

AWS DeepRacer 報酬関数の入力パラメータ

AWS DeepRacer の報酬関数は、辞書オブジェクトを入力として取ります。

```
def reward_function(params) :  
  
    reward = ...  
  
    return float(reward)
```

params 辞書オブジェクトには、次のキーと値のペアが含まれています。

```
{  
    "all_wheels_on_track": Boolean,          # flag to indicate if the agent is on the  
    track  
    "x": float,                             # agent's x-coordinate in meters  
    "y": float,                             # agent's y-coordinate in meters  
    "closest_objects": [int, int],          # zero-based indices of the two closest  
    objects to the agent's current position of (x, y).  
    "closest_waypoints": [int, int],        # indices of the two nearest waypoints.  
    "distance_from_center": float,          # distance in meters from the track center  
    "is_crashed": Boolean,                  # Boolean flag to indicate whether the agent  
    has crashed.  
    "is_left_of_center": Boolean,           # Flag to indicate if the agent is on the  
    left side to the track center or not.  
    "is_offtrack": Boolean,                 # Boolean flag to indicate whether the agent  
    has gone off track.  
    "is_reversed": Boolean,                 # flag to indicate if the agent is driving  
    clockwise (True) or counter clockwise (False).  
    "heading": float,                       # agent's yaw in degrees  
    "objects_distance": [float, ],          # list of the objects' distances in meters  
    between 0 and track_length in relation to the starting line.  
    "objects_heading": [float, ],           # list of the objects' headings in degrees  
    between -180 and 180.  
    "objects_left_of_center": [Boolean, ],  # list of Boolean flags indicating whether  
    elements' objects are left of the center (True) or not (False).  
    "objects_location": [(float, float),], # list of object locations [(x,y), ...].  
    "objects_speed": [float, ],             # list of the objects' speeds in meters per  
    second.  
    "progress": float,                       # percentage of track completed  
    "speed": float,                          # agent's speed in meters per second (m/s)  
    "steering_angle": float,                 # agent's steering angle in degrees  
    "steps": int,                            # number steps completed
```

```
"track_length": float,           # track length in meters.
"track_width": float,           # width of the track
"waypoints": [(float, float), ] # list of (x,y) as milestones along the
track center
}
```

入力パラメータに関するより詳細な技術リファレンスは以下のとおりです。

all_wheels_on_track

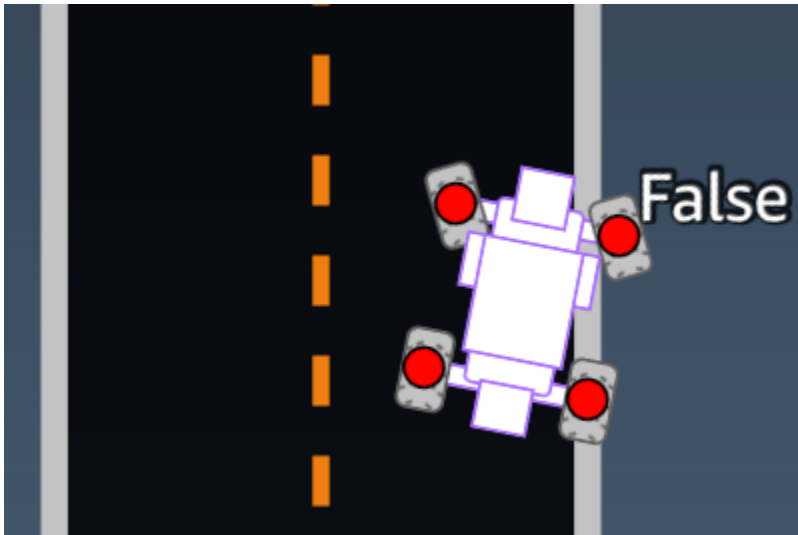
タイプ: Boolean

範囲: (True:False)

エージェントがトラック内にあるのかトラック外にあるのかを示す Boolean フラグ。ホイールのいずれかがトラックの境界線の外側にある場合は、トラック外 (False) です。すべてのホイールが2つのトラック境界の内側にある場合はトラック内 (True) です。次の図は、エージェントがトラック上にあることを示しています。



次の図は、エージェントがトラックから外れていることを示しています。



例: `all_wheels_on_track` パラメータを試用した報酬関数

```
def reward_function(params):  
    #####  
    ...  
    Example of using all_wheels_on_track and speed  
    ...  
  
    # Read input variables  
    all_wheels_on_track = params['all_wheels_on_track']  
    speed = params['speed']  
  
    # Set the speed threshold based your action space  
    SPEED_THRESHOLD = 1.0  
  
    if not all_wheels_on_track:  
        # Penalize if the car goes off track  
        reward = 1e-3  
    elif speed < SPEED_THRESHOLD:  
        # Penalize if the car goes too slow  
        reward = 0.5  
    else:  
        # High reward if the car stays on track and goes fast  
        reward = 1.0  
  
    return float(reward)
```

closest_waypoints

タイプ: [int, int]

範囲: [(0:Max-1), (1:Max-1)]

(x, y) のエージェントの現在位置に最も近い 2 つが隣接する waypoint のゼロベースのインデックス。距離は、エージェントの中心からのユークリッド距離によって測定されます。最初の要素は、エージェントの背後に最も近いウェイポイントを指し、2 番目の要素は、エージェントの前面にある最も近いウェイポイントを指します。Max は、ウェイポイントリストの長さです。[ウェイポイント](#) で示している図では、closest_waypoints は [16, 17] になります。

例: closest_waypoints パラメータを使用する報酬関数。

次の例の報酬関数は、waypoints と closest_waypoints、および heading を使用して即時報酬を計算する方法を示しています。

AWS DeepRacer は、数学、ランダム、NumPy、SciPy、Shapely のライブラリをサポートしています。1 つを使用するには、関数定義の上に、import *supported library*、インポートステートメントを追加します: def function_name(parameters)。

```
# Place import statement outside of function (supported libraries: math, random, numpy,
  scipy, and shapely)
# Example imports of available libraries
#
# import math
# import random
# import numpy
# import scipy
# import shapely

import math

def reward_function(params):
    #####
    ...
    Example of using waypoints and heading to make the car point in the right direction
    ...

    # Read input variables
    waypoints = params['waypoints']
    closest_waypoints = params['closest_waypoints']
```

```
heading = params['heading']

# Initialize the reward with typical value
reward = 1.0

# Calculate the direction of the center line based on the closest waypoints
next_point = waypoints[closest_waypoints[1]]
prev_point = waypoints[closest_waypoints[0]]

# Calculate the direction in radius, arctan2(dy, dx), the result is (-pi, pi) in
radians
track_direction = math.atan2(next_point[1] - prev_point[1], next_point[0] -
prev_point[0])
# Convert to degree
track_direction = math.degrees(track_direction)

# Calculate the difference between the track direction and the heading direction of
the car
direction_diff = abs(track_direction - heading)
if direction_diff > 180:
    direction_diff = 360 - direction_diff

# Penalize the reward if the difference is too large
DIRECTION_THRESHOLD = 10.0
if direction_diff > DIRECTION_THRESHOLD:
    reward *= 0.5

return float(reward)
```

closest_objects

タイプ: [int, int]

範囲: [(0:len(objects_location)-1), (0:len(objects_location)-1)]

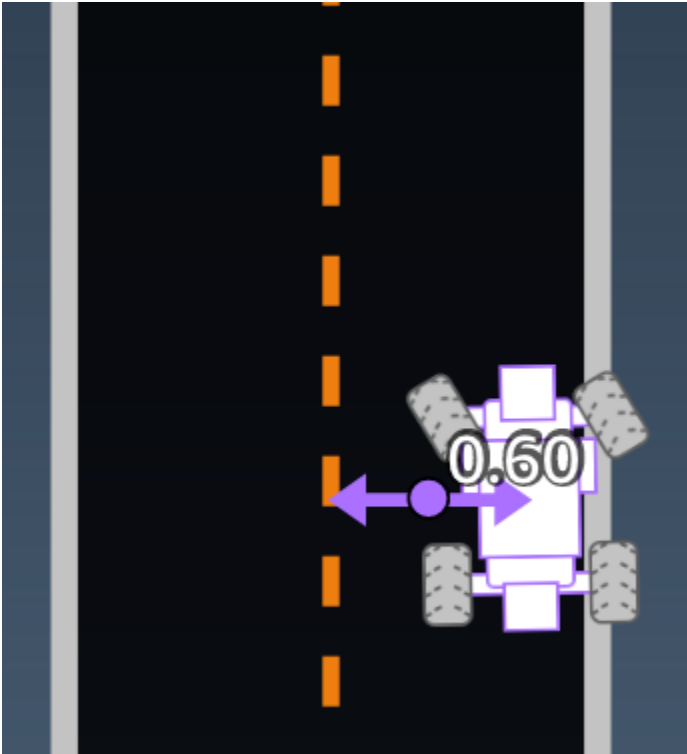
エージェントの現在の位置 (x, y) に最も近い 2 つのオブジェクトのゼロから始まるインデックス。最初のインデックスは、エージェントの背後にある最も近いオブジェクトを参照し、2 番目のインデックスは、エージェントの前にある最も近いオブジェクトを参照します。オブジェクトが 1 つしかない場合、両方のインデックスは 0 です。

distance_from_center

タイプ: float

範囲: 0:~track_width/2

エージェントの中心とトラックの中心との間のメートル単位の変位。観察可能な最大変位は、エージェントのいずれかの車輪がトラックの境界線の外側にあるときに発生し、トラックの境界線の幅に応じて、track_width の半分よりわずかに小さいまたは大きい場合があります。



例: `distance_from_center` パラメータを使用する報酬関数

```
def reward_function(params):  
    #####  
    ...  
    Example of using distance from the center  
    ...  
  
    # Read input variable  
    track_width = params['track_width']  
    distance_from_center = params['distance_from_center']  
  
    # Penalize if the car is too far away from the center  
    marker_1 = 0.1 * track_width
```

```
marker_2 = 0.5 * track_width

if distance_from_center <= marker_1:
    reward = 1.0
elif distance_from_center <= marker_2:
    reward = 0.5
else:
    reward = 1e-3 # likely crashed/ close to off track

return float(reward)
```

heading

タイプ: float

範囲: -180:+180

座標系の x 軸に対するエージェントの進行方向 (度単位)。



例: *heading* パラメータを使用する報酬関数

詳細については、「[closest_waypoints](#)」を参照してください。

is_crashed

タイプ: Boolean

範囲: (True:False)

エージェントが終了ステータスとして別のオブジェクトにクラッシュしたか (True)、否か (False) を示すブール型フラグ。

is_left_of_center

タイプ: Boolean

範囲: [True : False]

エージェントがトラックの中心より左側 (True) にあるのか右側 (False) にあるのかを示す Boolean フラグ。

is_offtrack

タイプ: Boolean

範囲: (True:False)

エージェントが終了ステータスとしてトラック外 (True) であるのかどうか (False) を示すブール型フラグ。

is_reversed

タイプ: Boolean

範囲: [True:False]

エージェントが時計回り (True) であるのか反時計回り (False) であるのかを示すブール型フラグ。

これは、エピソードごとに方向変更を有効にする場合に使用されます。

objects_distance

タイプ: [float, ...]

範囲: [(0:track_length), ...]

開始ラインに対する環境内のオブジェクト間の距離のリスト。i 番目の要素は、i 番目のオブジェクトと、トラックの中心線に沿った開始線間の距離をメートルで測定します。

Note

$\text{abs} | (\text{var1}) - (\text{var2})|$ = how close the car is to an object, WHEN $\text{var1} = [\text{"objects_distance"}]$ [index] and $\text{var2} = \text{params}[\text{"progress"}] * \text{params}[\text{"track_length"}]$
車両の前面に最も近いオブジェクトと車両の背後に最も近いオブジェクトのインデックスを取得するには、"closest_objects" パラメータを使用します。

objects_heading

タイプ: [float, ...]

範囲: [(-180:180), ...]

オブジェクトの見出しのリスト (度単位)。 i 番目の 要素は、 i 番目の オブジェクトの見出しを測定します。静止オブジェクトの場合、見出しは 0 です。ポット車両の場合、対応する要素の値は車両の見出し角度です。

objects_left_of_center

タイプ: [Boolean, ...]

範囲: [True|False, ...]

ブール型フラグのリスト。 i 番目の 要素の値は、 i 番目の オブジェクトがトラックセンターの左側 (True) か右側 (False) かを示します。

objects_location

タイプ: [(x,y), ...]

範囲: [(0:N,0:N), ...]

すべてのオブジェクトの場所のリスト。各場所は (x,y) のタプルです。

リストのサイズは、トラック上のオブジェクトの数と同じです。オブジェクトは、固定障害物、移動ポット車両である可能性があることに注意してください。

objects_speed

タイプ: [float, ...]

範囲: [(0:12.0), ...]

トラック上のオブジェクトの速度 (メートル/秒) のリスト。静止オブジェクトの場合、速度は 0 です。ボット車両の場合、値はトレーニングで設定した速度です。

プログレス

タイプ: float

範囲: 0:100

トラック完走の割合。

例: *progress* パラメータを使用する報酬関数

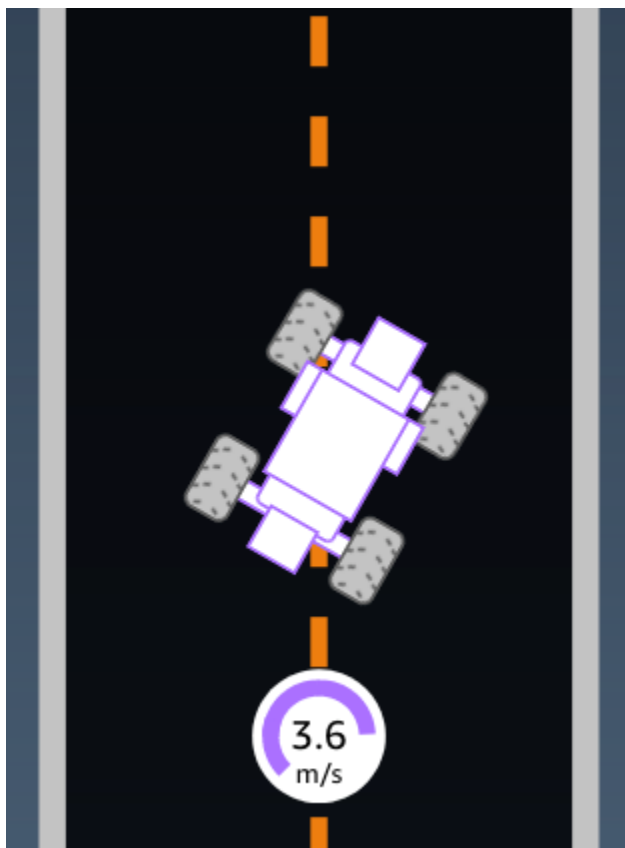
詳細については、「[ステップ](#)」を参照してください。

speed

タイプ: float

範囲: 0.0:5.0

エージェントの観測速度 (メートル/秒)。



例: *speed* パラメータを使用する報酬関数

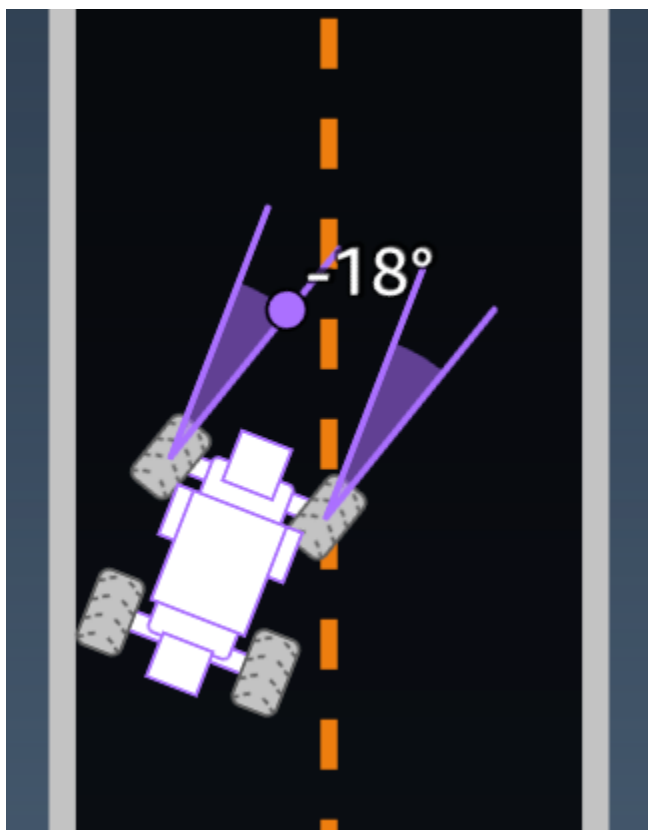
詳細については、「[all_wheels_on_track](#)」を参照してください。

steering_angle

タイプ: float

範囲: -30:30

エージェントの中心線からの前輪のステアリング角 (度単位)。負の記号 (-) は右へのステアリングを意味し、正の (+) 記号は左へのステアリングを意味します。次の図に示すように、エージェントの中心線はトラックの中心線と必ずしも平行ではありません。



例: *steering_angle* パラメータを使用する報酬関数

```
def reward_function(params):  
    ...  
    Example of using steering angle  
    ...
```

```

# Read input variable
abs_steering = abs(params['steering_angle']) # We don't care whether it is left or
right steering

# Initialize the reward with typical value
reward = 1.0

# Penalize if car steer too much to prevent zigzag
ABS_STEERING_THRESHOLD = 20.0
if abs_steering > ABS_STEERING_THRESHOLD:
    reward *= 0.8

return float(reward)

```

ステップ

タイプ: int

範囲: 0:N_{step}

完了したステップ数。ステップは、現在のポリシーに従ってエージェントがとるアクションに対応します。

例: *steps* パラメータを使用する報酬関数

```

def reward_function(params):
    #####
    ...
    Example of using steps and progress
    ...

    # Read input variable
    steps = params['steps']
    progress = params['progress']

    # Total num of steps we want the car to finish the lap, it will vary depends on the
    track length
    TOTAL_NUM_STEPS = 300

    # Initialize the reward with typical value
    reward = 1.0

    # Give additional reward if the car pass every 100 steps faster than expected

```

```
if (steps % 100) == 0 and progress > (steps / TOTAL_NUM_STEPS) * 100 :  
    reward += 10.0  
  
return float(reward)
```

track_length

タイプ: float

範囲: $[0: L_{\max}]$

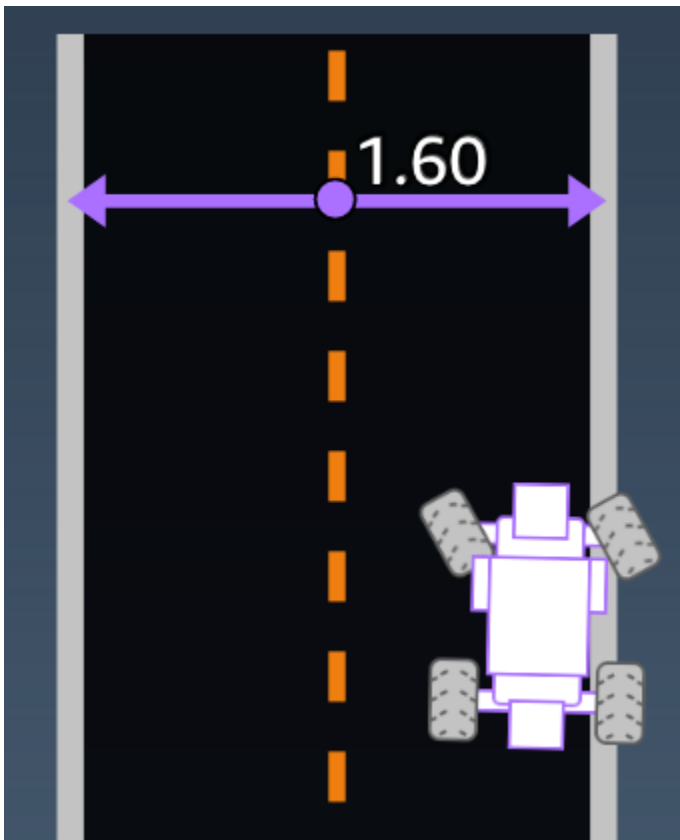
トラックの長さ (メートル単位)。 L_{\max} is track-dependent.

track_width

タイプ: float

範囲: $0: D_{\text{track}}$

トラックの幅 (メートル)。



例: `track_width` パラメータを使用する報酬関数

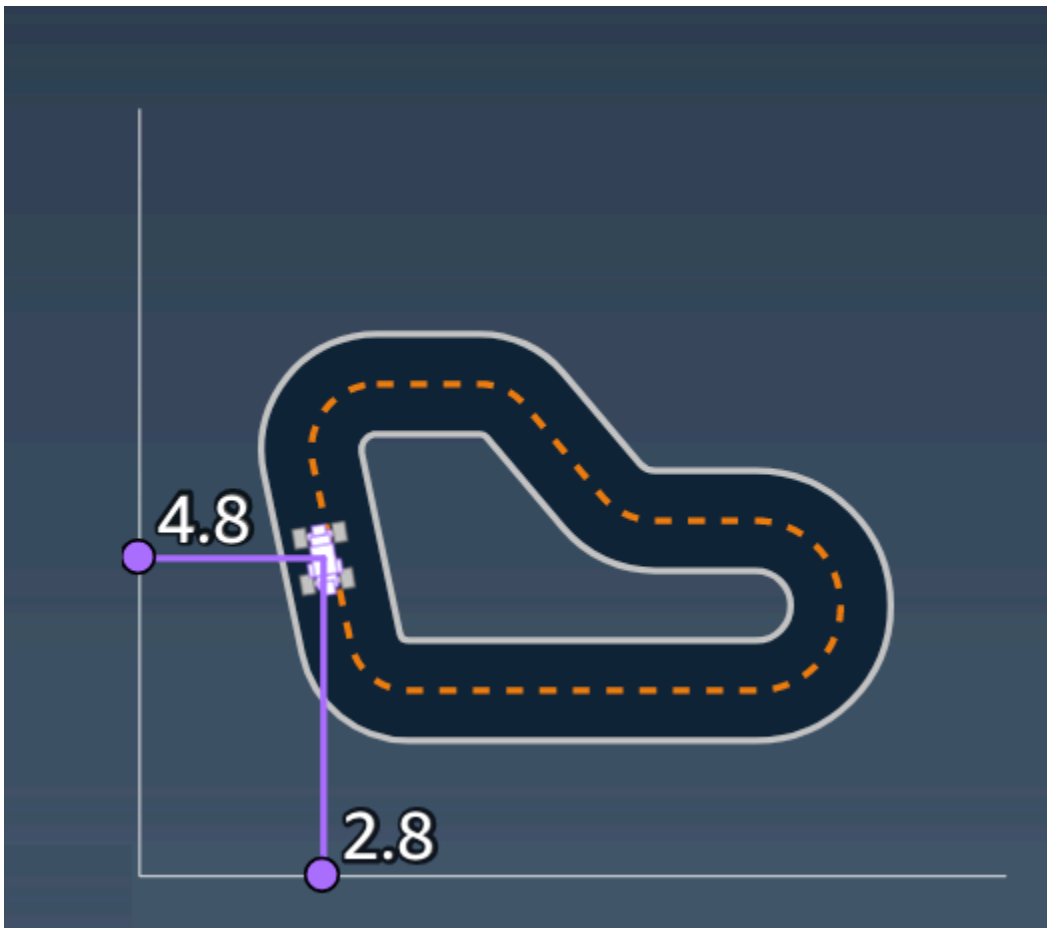
```
def reward_function(params):  
    #####  
    '''  
    Example of using track width  
    '''  
  
    # Read input variable  
    track_width = params['track_width']  
    distance_from_center = params['distance_from_center']  
  
    # Calculate the distance from each border  
    distance_from_border = 0.5 * track_width - distance_from_center  
  
    # Reward higher if the car stays inside the track borders  
    if distance_from_border >= 0.05:  
        reward = 1.0  
    else:  
        reward = 1e-3 # Low reward if too close to the border or goes off the track  
  
    return float(reward)
```

x、y

タイプ: float

範囲: 0:N

トラックを含むシミュレーション環境の x 軸と y 軸に沿ったエージェント中心の位置 (メートル単位)。原点は、シミュレーション環境の左下隅にあります。

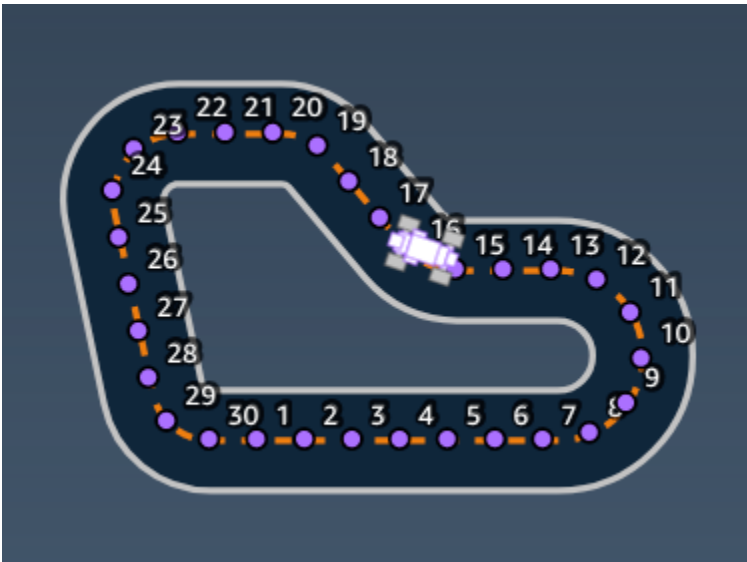


ウェイポイント

タイプ: [float, float] の list

範囲: $[[x_{w,0}, y_{w,0}] \dots [x_{w,Max-1}, y_{w,Max-1}]]$

トラックの中心に沿ったトラック依存 Max マイルストーンの順序付きリスト。各マイルストーンは、 $(x_{w,i}, y_{w,i})$ の座標で表されます。ループされたトラックの場合、最初と最後のウェイポイントは同じです。直線のトラックなどループされないトラックの場合、最初と最後のウェイポイントは異なります。



例 `waypoints` パラメータを使用する報酬関数

詳細については、「[closest_waypoints](#)」を参照してください。

AWS DeepRacer 報酬関数の例

以下に AWS DeepRacer の報酬関数のいくつかの例を示します。

トピック

- [例 1: タイムトライアルでセンターラインに従う](#)
- [例 2: タイムトライアルで 2 つの境界内に留まる](#)
- [例 3: タイムトライアルでのジグザグ運転の防止](#)
- [例 4: 静止している障害物や走行中の車両に衝突することなく、1 つの車線に留まること。](#)

例 1: タイムトライアルでセンターラインに従う

この例では、エージェントがセンターラインからどれだけ離れているかを調べ、トラックの中央に近いと高い報酬を与え、エージェントがセンターラインに密接に従うように促します。

```
def reward_function(params):  
    '''  
    Example of rewarding the agent to follow center line  
    '''  
  
    # Read input parameters
```

```
track_width = params['track_width']
distance_from_center = params['distance_from_center']

# Calculate 3 markers that are increasingly further away from the center line
marker_1 = 0.1 * track_width
marker_2 = 0.25 * track_width
marker_3 = 0.5 * track_width

# Give higher reward if the car is closer to center line and vice versa
if distance_from_center <= marker_1:
    reward = 1
elif distance_from_center <= marker_2:
    reward = 0.5
elif distance_from_center <= marker_3:
    reward = 0.1
else:
    reward = 1e-3 # likely crashed/ close to off track

return reward
```

例 2: タイムトライアルで 2 つの境界内に留まる

この例では、エージェントが境界内に留まる場合に高い報酬を与え、エージェントがラップを完了するための最良の経路を把握させます。プログラミングと理解は簡単ですが、収束に時間がかかる可能性があります。

```
def reward_function(params):
    ...
    Example of rewarding the agent to stay inside the two borders of the track
    ...

    # Read input parameters
    all_wheels_on_track = params['all_wheels_on_track']
    distance_from_center = params['distance_from_center']
    track_width = params['track_width']

    # Give a very low reward by default
    reward = 1e-3

    # Give a high reward if no wheels go off the track and
    # the car is somewhere in between the track borders
    if all_wheels_on_track and (0.5*track_width - distance_from_center) >= 0.05:
        reward = 1.0
```

```
# Always return a float value
return reward
```

例 3: タイムトライアルでのジグザグ運転の防止

この例では、エージェントがセンターラインに従うようにインセンティブを与えますが、操作が大きすぎると報酬が低くなり、ジグザグ運転を防ぐのに役立ちます。エージェントはシミュレーターでスムーズに運転することを学習すれば、実際の車両にデプロイされたときに同じ動作を維持できる可能性があります。

```
def reward_function(params):
    """
    Example of penalize steering, which helps mitigate zig-zag behaviors
    """

    # Read input parameters
    distance_from_center = params['distance_from_center']
    track_width = params['track_width']
    abs_steering = abs(params['steering_angle']) # Only need the absolute steering
angle

    # Calculate 3 marks that are farther and father away from the center line
    marker_1 = 0.1 * track_width
    marker_2 = 0.25 * track_width
    marker_3 = 0.5 * track_width

    # Give higher reward if the car is closer to center line and vice versa
    if distance_from_center <= marker_1:
        reward = 1.0
    elif distance_from_center <= marker_2:
        reward = 0.5
    elif distance_from_center <= marker_3:
        reward = 0.1
    else:
        reward = 1e-3 # likely crashed/ close to off track

    # Steering penalty threshold, change the number based on your action space setting
    ABS_STEERING_THRESHOLD = 15

    # Penalize reward if the car is steering too much
    if abs_steering > ABS_STEERING_THRESHOLD:
        reward *= 0.8
```

```
return float(reward)
```

例 4: 静止している障害物や走行中の車両に衝突することなく、1つの車線に留まること。

この報酬関数は、トラック境界の間に留まるエージェントに報酬を与え、前方のオブジェクトに近づきすぎたエージェントにペナルティを与えます。エージェントは、クラッシュを回避するために、車線から車線に移動することができます。報酬総額は、報酬とペナルティの加重合計です。この例では、衝突を回避するために、ペナルティをより重視しました。平均化の重みを変えて実験し、様々な行動の結果に対応できるようにトレーニングします。

```
import math
def reward_function(params):
    """
    Example of rewarding the agent to stay inside two borders
    and penalizing getting too close to the objects in front
    """
    all_wheels_on_track = params['all_wheels_on_track']
    distance_from_center = params['distance_from_center']
    track_width = params['track_width']
    objects_location = params['objects_location']
    agent_x = params['x']
    agent_y = params['y']
    _, next_object_index = params['closest_objects']
    objects_left_of_center = params['objects_left_of_center']
    is_left_of_center = params['is_left_of_center']
    # Initialize reward with a small number but not zero
    # because zero means off-track or crashed
    reward = 1e-3
    # Reward if the agent stays inside the two borders of the track
    if all_wheels_on_track and (0.5 * track_width - distance_from_center) >= 0.05:
        reward_lane = 1.0
    else:
        reward_lane = 1e-3
    # Penalize if the agent is too close to the next object
    reward_avoid = 1.0
    # Distance to the next object
    next_object_loc = objects_location[next_object_index]
    distance_closest_object = math.sqrt((agent_x - next_object_loc[0])**2 + (agent_y -
next_object_loc[1])**2)
```

```
# Decide if the agent and the next object is on the same lane
is_same_lane = objects_left_of_center[next_object_index] == is_left_of_center
if is_same_lane:
    if 0.5 <= distance_closest_object < 0.8:
        reward_avoid *= 0.5
    elif 0.3 <= distance_closest_object < 0.5:
        reward_avoid *= 0.2
    elif distance_closest_object < 0.3:
        reward_avoid = 1e-3 # Likely crashed
# Calculate reward by putting different weights on
# the two aspects above
reward += 1.0 * reward_lane + 4.0 * reward_avoid
return reward
```

AWS DeepRacer コンソールでモデルをインポートおよびエクスポートする

AWS DeepRacer モデルをインポートまたはエクスポートしなければならないシナリオがあります。雇用主が主催するイベントに参加したレーサーは、アクセス不可になるのを防ぐためにモデルをエクスポートできます。また、レース管理者は、参加者がイベント中にインポートして使用できるよう、事前にトレーニングされたモデルを提供できます。[お客様のモデル] ページを使用して、コンソールで AWS DeepRacer モデルをインポートおよびエクスポートします。

トピック

- [AWS DeepRacer モデルを Amazon S3 にコピーする](#)
- [AWS DeepRacer モデルをコンソールにインポートする](#)
- [トラブルシューティング](#)

AWS DeepRacer モデルを Amazon S3 にコピーする

AWS DeepRacer モデルを Amazon S3 にコピーするには

1. [AWS DeepRacer コンソール](#) にログインします。
2. ナビゲーションペインの [強化学習] で [お客様のモデル] を選択します。
3. モデル名の横にあるチェックボックスをチェックして、インポートするモデルを選択します。コンソールから Amazon S3 には 1 つのモデルしかコピーできません。
4. [アクション] ボタンのドロップダウンを選択し、[S3 へコピー] を選択します。

新しい [Amazon S3 へコピー] ページが開きます。

5. [Amazon S3 へコピー] ページで、Amazon S3 バケットドロップダウンセレクターを使用し、モデルのエクスポート先の Amazon S3 バケットを選択してください。AWS DeepRacer S3 バケットの名前には `deepacer` という言葉が含まれている必要があります。
 - 有効な Amazon S3 バケットを持っていない場合は、[新しいバケットを作成] を選択してバケットを作成します。ドロップダウンセレクターにより、バケット名に次の形式が入力されます `aws-deepracer-assets-XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX`。
6. S3 オブジェクトプレフィックスフィールドの Amazon S3 オブジェクトにオプションのフォルダプレフィックスを追加してください。

7. S3 バケットをセットアップしたら、含めたいアセットを選択してください。続行するには、少なくとも 1 つのアセットタイプを選択しなければなりません。
 - モデル: モデルフォルダーには、モデルのインポートのために必要なすべてのファイルが含まれています。
 - ログ: モデルのトレーニングログと評価ログをコピーします。このオプションには logs/、metrics/、そして sim-trace/ フォルダが含まれています。
 - ビデオ: このオプションはビデオフォルダを Amazon S3 バケットにコピーします。ビデオフォルダには、evaluation/ および training/ のフォルダが含まれます。これらのフォルダには、トップビュー、45 度アングルビュー、そしてトラック上の車の位置を示す、コンソールのオーバーレによる 45 度アングルビューのビデオが含まれています。
8. [コピー] をクリックした後、Amazon S3 データストレージの費用負担が発生することを知らせるポップアップが表示されます。条件に同意する場合、ポップアップの [コピー] ボタンを押してください。
9. コピー処理が開始されると、コンソールの [お客様のモデル] ページに戻ります。現在のステータスが、ページ上部のバナーに表示されます。エクスポート処理が完了すると、バナーにエクスポートが成功したことが表示されます。

モデルのインポートに必要なファイル

コンソール外でトレーニングされたモデルのモデルフォルダをアップロードするには、Amazon S3 ドキュメントの「[オブジェクトのアップロード](#)」ページの手順に従います。次の表には、モデルのインポートに必要なファイルのリストが含まれています。必要なファイルの一つでも欠けていると、モデルのインポートは失敗します。

AWS DeepRacer コンソールでトレーニングされたモデルは、フォルダ名形式 DAY/MONTH/YEAR/TIME GMT になっています。このサンプルモデルは 2023 年 11 月 30 日にエクスポートされ、フォルダ名は Thu, 30 Nov 2023 19:01:24 GMT です。このサンプルでは、フォルダはルートと呼ばれています。

モデルのインポートに必要なファイル

ファイル名	フォルダパス	説明
.coach_checkpoint	ルート/モデル/	コーチチェックポイントファイルには、インポートするために使用されたモデルチェッ

ファイル名	フォルダパス	説明
		クポイントのキーが含まれています。
ckpt ファイル	ルート/モデル/	チェックポイントファイルは、トレーニング中のさまざまな段階で取得されたモデルのウェイトのスナップショットです。それらには、ckpt.index、ckpt.data、そしてckpt.meta のファイルが含まれています。
model_metadata.json	ルート/	モデルメタデータファイルには、アクションスペース定義、センサー設定、トレーニングアルゴリズムの選択などの設定が含まれています。
reward_function.py	ルート/	モデルのトレーニングに使用される報酬関数を含む Python ファイル。

メトリクスファイルはモデルのインポートには必要ではありません。これらのファイルが含まれていない場合、モデルのトレーニングメトリクスと報酬グラフはコンソールに表示されません。

モデルインポートのためのオプションファイル

ファイル名	フォルダパス	説明
training_params.yaml	ルート/	training_params ファイルには、トラックや車両の情報、レーサーとモデルの名前、トレーニングアーティファクトのフォルダパスなどのト

ファイル名	フォルダパス	説明
		レーニングジョブデータが含まれています。
hyperparameters.json	ルート/IP/	バッチサイズ、ロスタイプ、学習率、エポック数など、モデルのハイパーパラメータ情報が含まれています。
training-*.json	ルート/メトリクス/トレーニング/	AWS DeepRacer コンソールでモデルのトレーニングメトリクスの視覚化のために使用されます。

AWS DeepRacer モデルをコンソールにインポートする


このセクションでは、AWS DeepRacer モデルをコンソールにインポートする手順について学んでいただきます。モデルをインポートする前に、モデルフォルダの Amazon S3 URL をコピーしなければなりません。

AWS DeepRacer Amazon S3 バケット URL をコピーします。

1. [Amazon S3 コンソール](#)にログインし、[バケット] ページに移動します。
2. バケット名のリンクを押して、最初の AWS DeepRacer モデル用に作成した Amazon S3 バケットを選択してください。AWS DeepRacer コンソールで作成される S3 バケットの形式は `aws-deepracer-assets-XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX` です。
3. AWS DeepRacer バケットの [オブジェクト] タブから:
 - a. [名前] フィールドにあるモデルオブジェクトの名前のリンクを押して、インポートするモデルを選択します。

サブフォルダのリストが表示されます。
 - b. フォルダー名の横にあるチェックボックスをチェックし、モデルフォルダーのルートを選択します。AWS DeepRacer コンソールでトレーニングされたモデルは、フォルダ名形式 `DAY/MONTH/YEAR/TIME GMT` になっています。

4. モデルフォルダのルートに移動したら、[S3 URL をコピー] ボタンを選択してください。[S3 URL のコピー]、[URL のコピー]、[開く]、[削除] などそれぞれのボタンは、モデル名の横にあるチェックボックスにチェックを入れるまでグレー表示されます。

 Note

S3 へのコピープロセス中に *my_model/version_2* などのプレフィックスを追加した場合、モデルフォルダのパスは `deep_racer_bucket/model_name/my_model/version_2/root/` になります。

AWS DeepRacer モデルをコンソールにインポートする

1. [AWS DeepRacer コンソール](#)で、[お客様のモデル] ページに移動します。
2. モデルコンテナの [モデルのインポート] ボタンを選択してください。
[モデルのインポート] ページが表示されます。
3. [インポート] セクションで:
 - インポートするモデルフォルダの Amazon S3 URL を入力します。Amazon S3 URL の形式は `s3://deep_racer_bucket/model_name/prefix/root` です。
4. 詳細セクションでは:
 - a. モデル名を入力します。
 - b. オプションロールの説明を任意で追加します。
 - c. マルチユーザーモードの管理者アカウントを使用している場合、モデルをインポートするユーザーをドロップダウンセレクターから選択します。
5. 画面の下部にある [インポート] ボタンを選択します。
6. インポートプロセスが開始されると、コンソールの [お客様のモデル] ページに戻ります。ページ上部に現在のステータスを示すバナーが表示され、モデルはモデルリストに [インポート中] のステータスで表示されます。インポート処理が完了すると、バナーにインポートの成功が表示され、モデルのステータスが [インポート中] から [準備完了] へと変わります。

トラブルシューティング

モデルのコピーエラー

何度か試したにもかかわらず、モデルをコピーできませんでした。モデルがまだ S3 バケット内にある場合は、[モデルエラー] テーブルからモデルを選択し、[更新] を選択し、[インポート] を選択して、モデルのインポートを再試行します。または、モデルのローカルコピーがある場合、Amazon S3 ドキュメントの「[オブジェクトのアップロード](#)」ページの手順に従って手動でインポートすることも可能です。

Amazon S3 バケットが存在しません。

このモデルが保存された S3 バケットが削除されていることにより、モデルをコピーできませんでした。モデルのコピーがある場合は、S3 バケット内にそれを deepracer と一緒に配置し、[AWS DeepRacer モデルをコンソールにインポートする](#) セクションの手順に従ってインポートをやり直してください。

Amazon S3 バケットにアクセスできない

このモデルが格納されている Amazon S3 バケットへのアクセス許可が変更されたため、モデルをコピーできませんでした。これは、AWS DeepRacer S3 または AWS DeepRacer サービスロールポリシーに対するアクセス権限を直接編集したなど、理由の 2 つの原因で発生します。AWS DeepRacer S3 バケットのアクセス権限を直接編集した場合は、「[Amazon S3 コンソールを使用してバケットポリシーを追加する](#)」ページの手順に従い、以下のポリシーを使用してバケットのアクセス権限を復元できます。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1586917903457",
      "Effect": "Allow",
      "Principal": {
        "Service": "deepracer.amazonaws.com"
      },
      "Action": [
        "s3:GetObjectAcl",

```

```
        "s3:GetObject",
        "s3:PutObject",
        "s3:PutObjectAcl"
    ],
    "Resource": [
        "arn:aws:s3:::your-bucket-name",
        "arn:aws:s3:::your-bucket-name/*"
    ]
}
]
```

バケットのアクセス権限を復元した後でモデルをインポートするには、[モデルエラー] テーブルからモデルを選択し、[更新] を選択します。[モデルのインポート] ページが表示されたら、[インポート] を選択します。

モデルファイルが存在しません。

モデルが Amazon S3 バケットから削除されているため、コピーできません。ファイルがまだ残っている場合は、それを AWS DeepRacer バケットに復元してから、[モデルエラー] テーブルからモデルを選択し、[更新] を選択します。[モデルのインポート] ページが表示されたら、[インポート] を選択します。または、モデルのローカルコピーがある場合、Amazon S3 ドキュメントの「[オブジェクトのアップロード](#)」ページの手順に従って手動でインポートすることが可能です。

Coach ファイルが存在しません。

Coach チェックポイントのメタデータが Amazon S3 バケットから削除されているため、モデルをコピーできません。ファイルがまだ残っている場合は、それを AWS DeepRacer バケットに復元してから、[モデルエラー] テーブルからモデルを選択し、[更新] を選択します。[モデルのインポート] ページが表示されたら、[インポート] を選択します。または、モデルのローカルコピーがある場合、Amazon S3 ドキュメントの「[オブジェクトのアップロード](#)」ページの手順に従って手動でインポートすることが可能です。

チェックポイントファイルが存在しません。

Coach チェックポイントのメタデータが Amazon S3 バケットから削除されているため、モデルをコピーできません。ファイルがまだ残っている場合は、それを AWS DeepRacer バケットに復元してから、[モデルエラー] テーブルからモデルを選択し、[更新] を選択します。[モデルのインポート] ページが表示されたら、[インポート] を選択します。または、モデルのローカルコピーがある場

合、Amazon S3 ドキュメントの「[オブジェクトのアップロード](#)」ページの手順に従って手動でインポートすることが可能です。

モデルファイルが大きすぎます。

モデルファイルがサービスで作成できる 1 GB のファイルサイズ制限を超えているため、ファイルは編集されました。このモデルはインポートされません。このメッセージを削除するには、モデルを[モデルエラー]から選択し、[削除]を選択します。

チェックポイントファイルが大きすぎます。

モデルファイルがサービスで作成できる 1 GB のファイルサイズ制限を超えているため、ファイルは編集されました。このモデルはインポートされません。このメッセージを削除するには、モデルを[モデルエラー]から選択し、[削除]を選択します。

メタデータファイルが大きすぎます。

YAML ファイルは、サービスが作成できる 10 MB のファイルサイズ制限を超えているため、ファイルが編集されました。このモデルはインポートされません。このメッセージを削除するには、モデルを[モデルエラー]から選択し、[削除]を選択します。

モデルが無効です。

モデルが編集されているため、モデルを検証できません。ファイルがまだ残っている場合は、それを AWS DeepRacer バケットに復元してから、[モデルエラー]テーブルからモデルを選択し、[更新]を選択します。[モデルのインポート]ページが表示されたら、[インポート]を選択します。

権限がないか正しくない

トレーニング時に AWS DeepRacer で利用可能だったアクセス権限が削除されたため、モデルをコピーできません。AWS DeepRacer に必要なアクセス権限を再作成することを許可するには、モデルを[モデルエラー]テーブルから選択し、[更新]を選択します。[モデルのインポート]ページが表示されたら、[インポート]を選択します。アクセス権限を再作成すると、AWS DeepRacer はモデルをコピーします。

AWS DeepRacer 車両を運転する

AWS DeepRacer シミュレーターで AWS DeepRacer モデルのトレーニングと評価を終えた後に、モデルを AWS DeepRacer 車両にデプロイすることができます。トラックで走行するように車両を設定、物理環境でモデルのパフォーマンスを評価できます。これは、実際の自動レースを模倣しています。

車両を初めて走行させる前に車両を設定し、ソフトウェアを更新させ、ドライブチェーン サブシステムをキャリブレートする必要があります。

物理的トラックで車両を走行させるには、トラックが必要です。詳細については、「[物理的なトラックを構築する](#)」を参照してください。

トピック

- [AWS DeepRacer 車両について理解する](#)
- [AWS DeepRacer 車両用 Wi-Fi ネットワークを選択する](#)
- [AWS DeepRacer 車両のデバイスコンソールを起動する](#)
- [AWS DeepRacer 車両をキャリブレートする](#)
- [AWS DeepRacer 車両にモデルをアップロードする](#)
- [AWS DeepRacer 車両を運転する](#)
- [AWS DeepRacer 車両の設定を確認、管理する](#)
- [AWS DeepRacer 車両のログを表示する](#)

AWS DeepRacer 車両について理解する

AWS DeepRacer 車両は機械学習対応、バッテリー駆動、Wi-Fi 接続型の 4 輪駆動の 1/18 スケールモデルで、フロントマウントの 4 メガピクセルカメラと Ubuntu ベースのコンピューティングモジュールが搭載されています。

コンピューティングモジュールの強化学習モデルに基づく推論を実行することで、車両は自律走行できます。また、強化学習モデルをデプロイせず、車両を手動で走行させることもできます。AWS DeepRacer の車両をまだ入手していない場合は、[こちらから注文](#)できます。

AWS DeepRacer の車両は、ブラシ付きモーターで駆動しています。走行速度は、モーターの回転速度を制御する電圧レギュレータによって制御されます。ステアリングシステムを操作する[サーボ機構 \(サーボ\)](#)は、AWS DeepRacer の車両シャーシの黒いカバーによって保護されています。

トピック

- [AWS DeepRacer 車両を検査する](#)
- [AWS DeepRacer バッテリーを充電して取り付ける](#)
- [AWS DeepRacer コンピューティングモジュールをテストする](#)
- [AWS DeepRacer 車両をオフにする](#)
- [AWS DeepRacer 車両の LED インジケータ](#)
- [AWS DeepRacer デバイスのスペアパーツ](#)

AWS DeepRacer 車両を検査する

AWS DeepRacer の車両ボックスを開くと、以下のコンポーネントとアクセサリがあるはずです。



コンポーネント	コメント
車両シャーシ [1]	車両の走行体験をキャプチャするためのフロントマウントカメラと自動運転のためのコンピューティングモジュールを搭載しています。

コンポーネント	コメント
車両ボディシエル [2]	車両を設定するときには取り外します。
マイクロ USB to USB-A ケーブル [3]	USB-OTG 機能をサポートするために使用します。
コンピューティングバッテリー [4]	ダウンロードした AWS DeepRacer 強化学習モデル上で実行するコンピューティングモジュールに電力を供給するために使用します。
コンピューティングバッテリーコネクタケーブル [5]	この USB-C to USB-C ケーブルは、コンピューティングモジュールとバッテリーを接続するために使用します。Dell のコンピューティングバッテリーをお持ちの場合、このケーブルは長いものになります。
電源ケーブル [6a]	電源アダプターをコンセントに接続する場合に使用します。
電源アダプター [6b]	これを使用して、コンピューティングバッテリーとコンピューティングモジュールを充電します。
ピン (スペアパーツ) [7]	コンピューティングモジュールを車両のシャーシに固定します。これらは予備です。
車両バッテリー [8]	モーターに電力を供給する 7.4v LiPo バッテリーパック。
車両バッテリー充電アダプター [9a]	車両ドライブチェーンに電力を供給する車両バッテリーの充電に使用します。

コンポーネント	コメント
車両バッテリー充電ケーブル [9b]	車両バッテリーチャージャーを電源コンセントに接続するために使用します。
バッテリーロック解除ケーブル [10]	バッテリーがロックアウト状態になった場合に使用します。

AWS DeepRacer の車両をセットアップするには、以下のアイテムも用意する必要があります。

- USB ポートを備え、インターネットにアクセスできるコンピュータ。
- インターネットに接続されている Wi-Fi ネットワーク。
- AWS アカウント。

ここでは、[次のセクション](#)の指示に従って車両バッテリーとモバイルバッテリーが充電されていることを確認します。

AWS DeepRacer バッテリーを充電して取り付ける

AWS DeepRacer 車両には、車両バッテリーとコンピューティングモジュールパワーバンクの 2 つの電源があります。

モバイルバッテリーはコンピューティングモジュールを稼働状態で保持します。このコンピューティングモジュールは、Wi-Fi 接続を維持し、デプロイされた AWS DeepRacer モデルに対して推論を実行し、車両がアクションを実行するためのコマンドを発行します。

車両バッテリーは、車両を駆動するモーターに電力を供給します。2 つのケーブルセットがあります。赤と黒の 2 線式ケーブルは、車両の ESC への接続に使用し、青 (または黒)、白、赤の 3 線式ケーブルは、充電器への接続に使用します。走行の場合、2 線式ケーブルのみを車両に接続してください。

完全に充電された後、バッテリーの電圧はバッテリーが放電するとともに低下します。電圧が低下すると、使用可能なトルクも低下します。その結果、速度設定が同じでも、トラックでの速度は遅くなります。バッテリーが完全に空になると、車両は停止します。通常の条件下での自動走行の場合、バッテリーは通常 15 ~ 25 分間持続します。一貫した動作を保つには、15 分ごとにバッテリーを充電することをお勧めします。

車両バッテリーとモバイルバッテリーを取り付けて充電するには、次のステップを実行します。

1. AWS DeepRacer 車両シエルを取り外します。
2. 4本の車両フレームピンを取り外します。ワイヤを接続したまま、慎重に車両シャーシを持ち上げます。
3. 車両バッテリーを充電して取り付けるには、以下の手順を実行します。
 - a. バッテリーを充電するには、まず、バッテリーの3線式ケーブルを充電器に接続します。次に、充電器を電源アダプターに接続し、電源アダプターをコンセントに接続します。または、USBケーブルを使用してバッテリーを充電する場合は、充電器に接続したUSBケーブルをUSBポートに接続します。

同梱の充電器を使用して車両バッテリーを充電する方法の図解については、「[the section called “車両のドライブモジュールのバッテリーを充電する方法”](#)」を参照してください。

- b. バッテリーの充電が完了したら、車両バッテリーの2線式ケーブルを車両の黒と赤のケーブルコネクタに接続します。
- c. 車両バッテリーを固定するため、付属のストラップでバッテリーを車両シャーシの下に結び付けます。

すべてのケーブルが車両内に入っていることを確認してください。

- d. 車両バッテリーが充電されているかどうかを確認するには、次の操作を行います。
 - i. 車両の電源スイッチをスライドさせて車両をオンにします。
 - ii. 短いビープ音が2回聞こえるかどうかを確認します。


ビープ音が聞こえない場合、車両は充電されていません。バッテリーを車両から取り外し、上記のステップ1を繰り返してバッテリーを再充電します。

- iii. 車両を使用していないときは、車両の電源スイッチを後ろにスライドさせて、車両バッテリーをオフにします。

4. モバイルバッテリーの充電レベルを確認するには、以下の操作を行います。
 - a. モバイルバッテリーの電源ボタンを押します。
 - b. 電源ボタンの横にある4つのLEDライトを確認して充電レベルを確認します。

4つのLEDライトがすべて点灯している場合、モバイルバッテリーは完全に充電されています。LEDライトが点灯しているが1つも点灯していない場合、モバイルバッテリーの充電が必要です。

- c. モバイルバッテリーを充電するには、充電アダプタの USB C プラグをモバイルバッテリーの USB C ポートに差し込みます。モバイルバッテリーが完全に充電されるまでには時間がかかります。充電されたら、[ステップ 4] を繰り返して、モバイルバッテリーが完全に充電されていることを確認します。
5. モバイルバッテリーを取り付けるには、次の作業を行います。
 - a. 電源ボタンと USB C ポートを車両の後ろに向けてモバイルバッテリーをホルダーに差し込みます。
 - b. ストラップを使用して、車両モバイルバッテリーをシャーシにしっかり結び付けます。

 Note

このステップでモバイルバッテリーをコンピューティングモジュールに接続しないでください。

AWS DeepRacer コンピューティングモジュールをテストする

コンピューティングモジュールをテストして、正常に開始できることを確認します。外部電源を使用してモジュールをテストするには、次の手順に従います。

車両のコンピューティングモジュールをテストするには

1. コンピューティングモジュールを電源に接続します。電源コードを電源アダプタに接続し、電源コードを電源コンセントに差し込んで、電源アダプタ USB C プラグをコンピューティングモジュールの USB C ポートに差し込みます。
2. コンピューティングモジュールの電源ボタンを押して、車両のコンピューティングモジュールをオンにします。
3. コンピューティングモジュールのステータスを確認するには、LED ライトが次のように表示されていることを確認します。
 - 青色で点灯

コンピューティングモジュールが開始され、指定された Wi-Fi に接続されていて、使用する準備ができています。

この状態では、HDMI ケーブル、USB マウスと USB キーボードを使用してコンピューティングモジュールをモニターに接続し、ログインできます。初回ログインの場合は、ユーザー名とパスワードの両方に対して `deepracer` を使います。以降のログインでは、パスワードをリセットするよう求められます。セキュリティ上の理由から、新しいパスワードには強力なパスワードフレーズを選択してください。

- 赤色で点滅

コンピューティングモジュールは設定モードです。

- 黄色で点灯

コンピューティングモジュールは初期化中です。

- 赤色で点灯

コンピューティングモジュールは Wi-Fi ネットワークへの接続に失敗しました。

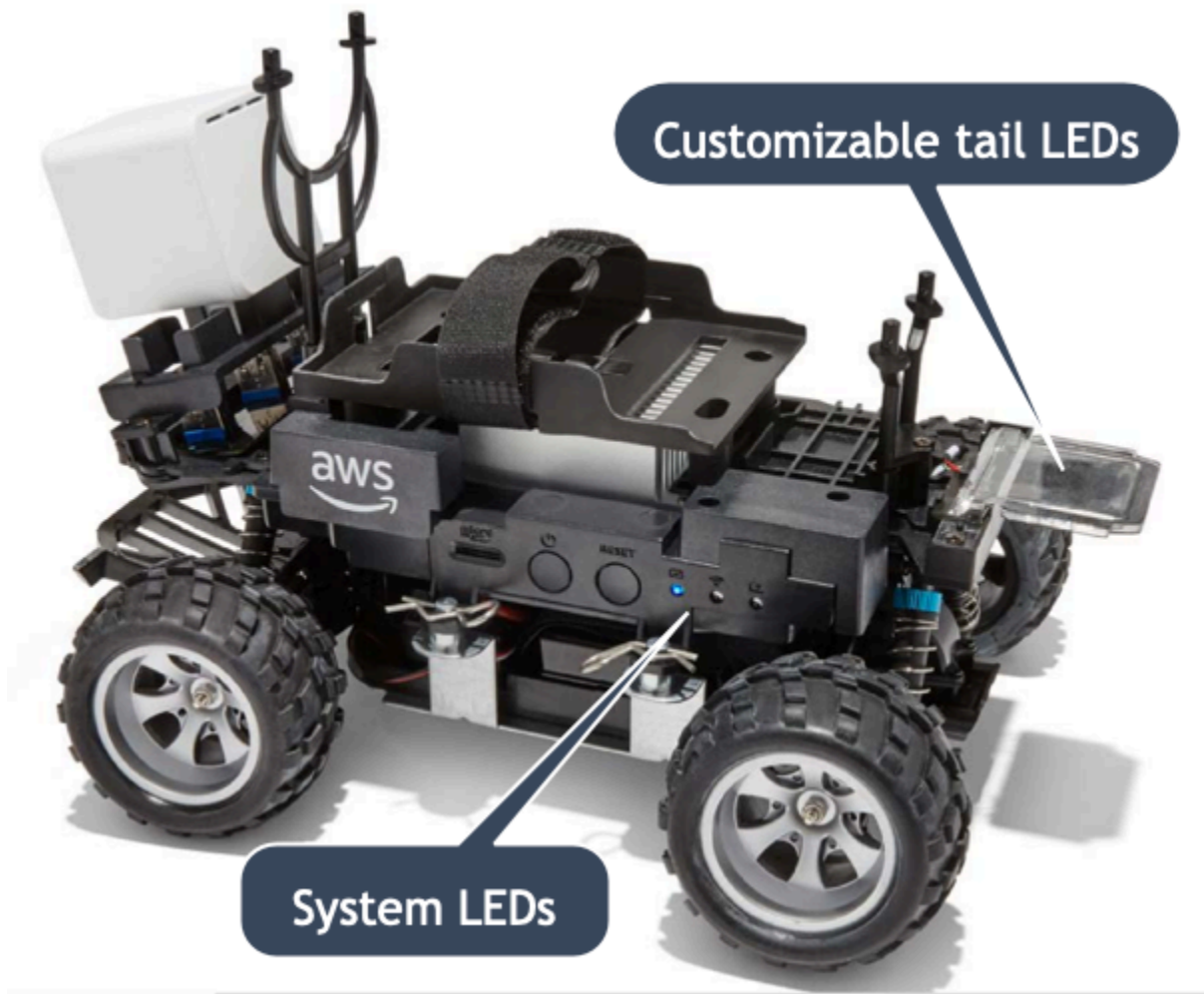
4. テストが完了したら、コンピューティングモジュールの電源ボタンを押してオフにし、外部電源から外します。

AWS DeepRacer 車両をオフにする

AWS DeepRacer 車両をオフにするには、車両から外部電源を取り外します。電源インジケータがオフになるまでデバイスの電源ボタンを押すこともできます。

AWS DeepRacer 車両の LED インジケータ

あなたの AWS DeepRacer 車両には、車両のステータスと車両の視覚的な識別 (カスタマイズ可能) のための LED インジケータが 2 セットあります。



詳細については、以下で説明します。

トピック

- [AWS DeepRacer 車両のシステム LED インジケータ](#)
- [AWS DeepRacer 車両識別 LED](#)

AWS DeepRacer 車両のシステム LED インジケータ

AWS DeepRacer 車両システムの LED インジケータは、貴方の前にある車両の前方から見て、車両のシャーシ左側にあります。

3つのシステム LED は、[リセット] ボタンの後に配置されています。最初の LED (視野の左側) は、システム電源の状態を示します。2番目 (中間) の LED は、将来の使用のために予約されています。最後 (右側) の LED は、Wi-Fi 接続のステータスを示します。

LED のタイプ	色	ステータス
電源	オフ	電源が供給されていません。
	黄色の点滅	BIOS と OS をロード中です。
	黄色の点灯	OS がロードされました。
	青色の点灯	アプリケーションが実行中です。
	青で点滅	ソフトウェアの更新が進行中です。
	赤色の点灯	システムまたはアプリケーションの起動中にエラーが発生しました。
Wi-Fi	オフ	Wi-Fi 接続がありません。
	青で点滅	車両が Wi-Fi ネットワークに接続中です。
	赤色で 2 秒間点灯後にオフ	Wi-Fi 接続に失敗しました。
	青色の点灯	Wi-Fi 接続が確立されました。

AWS DeepRacer 車両識別 LED

AWS DeepRacer 車両のカスタム LED は車両後部にあります。これらは、複数の車両が存在する場合に、レースで自分の車両を識別しやすくするために使用されます。AWS DeepRacer デバイスコンソールを使用して、任意の[サポートされている色を設定](#)できます。

AWS DeepRacer デバイスのスペアパーツ

Note

AWS DeepRacer デバイスでは、[WLT toys A949 および A979](#) リモートコントロール (RC) カーのシャーシを使用しています。お使いの AWS DeepRacer デバイスで入手可能なパーツの最新リストを参照するには、「[AWS DeepRacer ストアフロント](#)」にアクセスしてください。

AWS DeepRacer デバイスのスペアパーツ

パーツ	名前
	スペアコンピューティングバッテリー

パーツ	名前
	<p><u>スペアコンピューティングバッテリー</u></p>

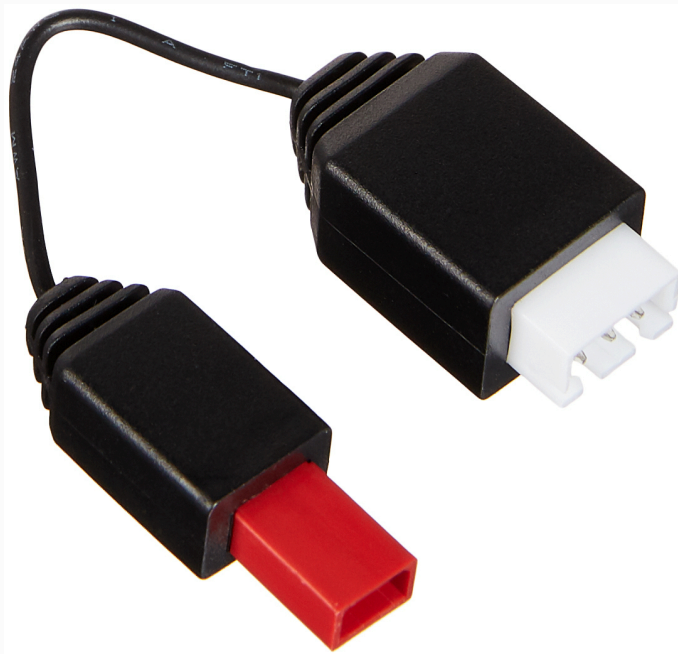
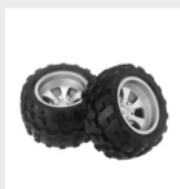
パーツ

名前

[スペアコンピューティングバッテリー](#)

パーツ

名前



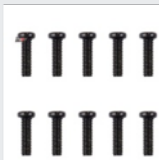

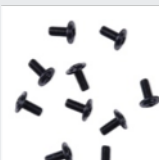
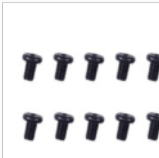
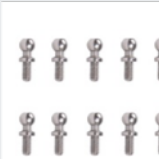
リチウム電池 7.4 V 1100 mAhAWS DeepRacer カーバッテリーロック解除ケーブルタイヤフロントバンパー

パーツ	名前
	サスペンションアーム
	プルロッド
	Cスタイルシート
	トランスミッションシャフト
	丸頭ねじ、M2x17.5 mm
	シャーシ自動車下部
	回転シート

パーツ	名前
	リアサスペンションフレーム
	金属六角コンバイナセット
	ギアボックスシェル
	ディファレンシャルボックスケース
	ディファレンシャルドライブカップ
	フロントリアシifter
	サーボシート

パーツ	名前
	中央駆動軸
	ショックフレーム
	サーボアーム
	差動機構
	減速ギア
	モーターベース
	17g ステアリングエンジン

パーツ	名前
	モーターのスクリューガスケット、据え付け
	390 モーター
	六角コネクタ 4x8x3 mm
	六角コネクタ 8x12x3.5 mm
	ボールベアリング 7x11x3 mm
	ボールベアリング 8x12x3.5 mm
	中間車軸ディスクプレート

パーツ	名前
	ねじ 2.6x6 mm
	ねじ 2x7 mm
	ねじ 2.5x8 mm
	ねじ 2x16 mm
	ねじ 2.5x6x6 mm
	ねじ M3x5 mm
	ボールねじ 10.8x4 mm

パーツ	名前
	ねじ 2x6 mm
	ねじ 2x9.5 mm
	M3 ロックナット
	車軸ヒンジピン
	ドライブシャフト
	スイングアームピン
	ねじ 2x29KM

パーツ	名前
	<u>ヘアピン</u>
	<u>フロントショックアブソーバー</u>
	<u>チャージャー</u>
	<u>金属モーターピニオンギア</u>
	<u>リアショックアブソーバー</u>
	<u>ESC</u>

AWS DeepRacer 車両用 Wi-Fi ネットワークを選択する

AWS DeepRacer 車両を初めて開いたときは、設定して Wi-Fi ネットワークに接続する必要があります。この設定を完了後、車両のソフトウェアを更新し、車両のデバイスのコンソールにアクセスするための IP アドレスを取得します。

このセクションでは、以下のタスクを実行する手順について説明します。

- ラップトップまたはデスクトップコンピュータを車両に接続します。
- 車両の Wi-Fi 接続を設定します。
- 車両のソフトウェアを更新します。
- 車両の IP アドレスを取得します。
- 車両をテストドライブします。

ラップトップまたはデスクトップコンピュータを使用して設定タスクを実行します。この設定用コンピュータは、Ubuntu オペレーティングシステムが実行されている車両のコンピューティングモジュールとの混同を避けるため、ユーザーのコンピュータと呼びます。

Wi-Fi 接続の初回の設定後、同じ手順で別の Wi-Fi ネットワークを選択できます。

Note

AWS DeepRacer では、ユーザーのサインインに有効な [キャプチャ](#) 検証を必要とする Wi-Fi ネットワークはサポートされていません。

トピック

- [AWS DeepRacer 車両の Wi-Fi 接続を設定する準備を整える](#)
- [Wi-Fi 接続の設定と AWS DeepRacer 車両のソフトウェアアップデート](#)

AWS DeepRacer 車両の Wi-Fi 接続を設定する準備を整える

車両の Wi-Fi 接続を設定するには、付属の USB-USB C ケーブルを使用して、ラップトップまたはデスクトップコンピュータを車両のコンピューティングモジュールに接続します。

ユーザーのコンピュータを車両のコンピューティングモジュールに接続するには、次のステップを実行します。

1. デバイスを接続する前に、コンピューターが Wi-Fi から切断されていることを確認してください。
2. USB-USB C ケーブルの USB 側をユーザーのコンピューターの USB ポートに挿入します。
3. ケーブルの USB C 側を車両の USB C ポートに挿入します。

これで、車両の Wi-Fi 接続に進む準備ができました。

Wi-Fi 接続の設定と AWS DeepRacer 車両のソフトウェアアップデート

以下の手順を実行して Wi-Fi 接続を設定する前に、[the section called “ Wi-Fi を設定する準備を整える ”](#) の手順を完了していることを確認してください。

1. 車両の下部を見て、[ホスト名] の下に印刷されたパスワードをメモします。設定を実行するには、デバイス管理コンソールにログインする必要があります。
2. コンピュータで、<https://deepracer.aws> に移動して、車両のデバイス管理コンソールを起動します。
3. 接続がプライベートではない、または保護されていないというメッセージが表示されたら、次のいずれかを実行します。
 - a. Chrome で、[詳細設定] を選択し、[**<device_console_ip_address>** (unsafe) (<device_console_ip_address> (unsafe) に進む)] を選択します。
 - b. Safari で、[詳細] を選択し、[このウェブサイトにはアクセス] リンクに従って、[ウェブサイトにはアクセス] を選択します。証明書の信頼設定を更新するためにパスワードを求められたら、パスワードを入力して [設定の更新] を選択します。
 - c. Opera では、無効な証明書の警告が表示されたら、[構わず続行する] を選択します。
 - d. Edge では、[詳細] を選択し、[Web ページに移動 (非推奨)] を選択します。
 - e. Firefox では、[詳細]、[例外の追加]、[セキュリティ例外を確認] の順に選択します。
4. [AWS DeepRacer 車両のロック解除] の下に [ステップ 1] でメモしたパスワードを入力し、[車両にアクセスする] を選択します。
5. [車両で Wi-Fi ネットワークに接続する] ペインで [Wi-Fi ネットワーク名 (SSID)] ドロップダウンメニューから Wi-Fi ネットワーク名を選択し、[Wi-Fi パスワード] に Wi-Fi ネットワークのパスワードを入力して、[接続] を選択します。
6. Wi-Fi 接続ステータスが [Wi-Fi ネットワークに接続中...] から [接続済み] になるまで待ちます。その後で、[次へ] を選択します。

7. ソフトウェアの更新が必要な場合、[ソフトウェア更新] ペインで、付属の電源ケーブルと電源アダプタで車両のコンピューティングモジュールをオンにして、[ソフトウェア更新をインストール] を選択します。

外部電源で車両をオンにすると、コンピューティングモジュールのバッテリーがなくなった場合にソフトウェア更新が中断するのを避けるのに役立ちます。
8. ソフトウェア更新ステータスが [ソフトウェア更新インストール中] から [ソフトウェア更新が正常にインストールされました] になるまで待ちます。
9. [Wi-Fi ネットワークの詳細] の下に表示される IP アドレスをメモしておきます。最初のセットアップの後および後で Wi-Fi ネットワーク設定を変更する場合に、車両のデバイス管理コンソールを開くために必要になります。

AWS DeepRacer 車両のデバイスコンソールを起動する

車両の Wi-Fi を設定し、必要なソフトウェア更新をインストールした後、車両のデバイスコンソールを開いて車両のネットワーク接続が機能しているかどうかを確認する必要があります。その後、デバイスコンソールを起動して、車両の他の設定を確認、キャリブレート、管理できます。このプロセスでは、車両の IP アドレスを使用して車両のデバイスコンソールにサインインする必要があります。

デバイス管理コンソールは車両でホストされており、「[Wi-Fi 設定](#)」セクションの末尾で取得した IP アドレスを使用してアクセスします。

Wi-Fi 接続を介して AWS DeepRacer 車両のデバイスコンソールにアクセスするには

1. 車両のデバイスコンソールにアクセスするには、コンピュータ、タブレット、またはスマートフォンのウェブブラウザを開き、アドレスバーに車両の IP アドレスを入力します。

この IP アドレスは、[車両の Wi-Fi 接続を設定](#)するときに取得できます。ここでの説明には、例として 10.92.206.61 を使用します。

接続がプライベートではない、または保護されていないという警告が表示された場合は、メッセージを無視して、デバイスコンソールに接続します。

2. [AWS DeepRacer 車両をロック解除] の下にある [パスワード] にコンソールのパスワードを入力し、[車両にアクセスする] を選択します。



Unlock your AWS DeepRacer vehicle

The default AWS DeepRacer password can be found printed on the bottom of your vehicle.

Password

Access vehicle

[Forgot password](#)

デフォルトのパスワードは車両の下側 ([ホスト名] の下) に印刷されています。

- サインインすると、デバイスコンソールのホームページが次のように表示されます。

The screenshot displays the 'Control vehicle' interface. On the left is a sidebar with the following items: 'Control vehicle' (highlighted), 'Models', 'Calibration', 'Settings', 'Logs', 'Build a track', 'Train a model', 'IP: 192.168.15.9', 'IP: 10.6.24.122', 'Vehicle battery level: Green', and a 'Logout' button. The main area is titled 'Control vehicle' and includes a 'Full screen' button. It features a 'Camera stream' showing a room with a desk and chair, and a 'Controls' section with 'Autonomous driving' selected, a 'Manual driving' option, a 'Select a model' dropdown, a 'Maximum speed' slider at 50%, and 'Start vehicle' and 'Stop vehicle' buttons. A 'Video stream' toggle is visible at the bottom left of the main area.

これで、車両のキャリブレーションと操作を行う準備ができました。初めて車両を運転する場合は、この時点で[車両の調整](#)に進みます。

AWS DeepRacer 車両をキャリブレートする

最高のパフォーマンスを達成するためには、お客様の AWS DeepRacer 車両の物理的なパーツをキャリブレートすることが不可欠です。キャリブレートされていない車両を使用すると、モデルのテスト時に不確実性が増す可能性があります。車両のパフォーマンスが最適ではない場合は、深層学習モデルのコードを調整するだけの方がよいでしょう。ただし、根本的な原因が機械的なものがある場合は、車両のパフォーマンスを向上させることはできません。キャリブレーションによってメカニズムを調整してください。

お客様の AWS DeepRacer の車両をキャリブレートするには、車両の電子制御システム (ECS) とそのサーボ機構 (サーボ) の[デューティサイクル](#)の範囲をそれぞれ設定します。サーボと ECS はどちらも、車両のコンピューティングモジュールからの制御入力として[パルス幅変調 \(PWM\)](#) 信号を受け入れます。コンピューティングモジュールは、PWM 信号のデューティサイクルを変更することによって、車両の速度とステアリング角の両方を調整します。

最大速度とステアリング角度は、アクションスペースのスパンを定義します。シミュレーションでトレーニング中に最大速度と最大ステアリング角を指定できます。実世界のトラックを走行するためにトレーニングされたモデルを AWS DeepRacer 車両にデプロイする場合は、シミュレーションでトレーニングで使用したものと一致するように車両の最大速度とステアリング角度をキャリブレートする必要があります。

実世界のエクスペリエンスとシミュレートされたエクスペリエンスを確実に一致させるには、シミュレーションと実世界の間の最大速度と最大ステアリング角度を一致させるように車両をキャリブレートする必要があります。一般的に、このキャリブレーションを実行するには 2 つの方法があります。

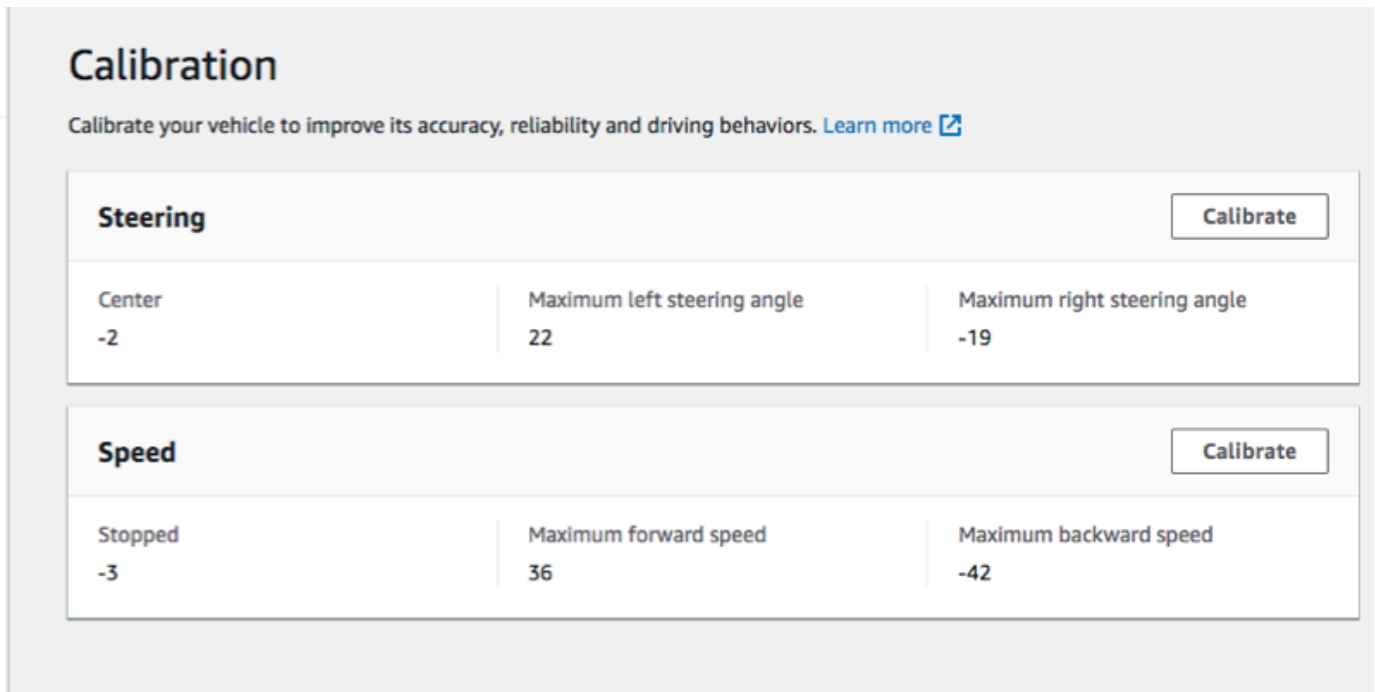
- トレーニングでアクションスペースを定義し、設定に合うように実際の車両をキャリブレートします。
- 車両の実際の性能を測定し、シミュレーションでアクションスペースの設定を変更してください。

堅牢なモデルは、シミュレーションと実世界との間の特定の違いに対応できます。ただし、最良の結果を見つけるために両方のアプローチを試して、反復する必要があります。

キャリブレーションを開始する前に、コンピューティングモジュールの電源を入れてください。起動して電源 LED が青色に点灯したら、車両のバッテリーをオンにします。短いビープ音が 2 回鳴り、長いビープ音が 1 回鳴ったら、キャリブレーションを続行できます。

トレーニングの設定に合わせて AWS DeepRacer 車両をキャリブレートする方法:

1. [これらの手順](#)に従って車両にアクセスし、デバイスコントロールコンソールを開きます。
2. メインのナビゲーションペインで、[キャリブレーション] を選択します。



Calibration			
Calibrate your vehicle to improve its accuracy, reliability and driving behaviors. Learn more			
Steering			Calibrate
Center	Maximum left steering angle	Maximum right steering angle	
-2	22	-19	
Speed			Calibrate
Stopped	Maximum forward speed	Maximum backward speed	
-3	36	-42	

3. リポジトリの [キャリブレーション] のページで [操縦] 内にある [キャリブレートする] を選択します。以下の手順に従って、車両の最大ステアリング角度を調整します。
 - a. ステアリングのキャリブレーション中に車輪を見ることができる地面または他の硬い表面に車両を置きます。[次へ] をクリックします。

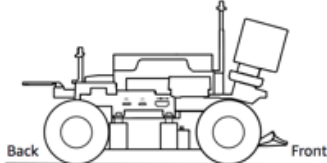
Calibration > Calibrate steering angle

Step 1
Set your vehicle on the ground

Calibrate steering angle

Set vehicle on the ground

Place your vehicle on the ground or other hard surface within eyesight. You must be able to see the wheels during steering calibration.



Cancel Next

トラックで車両を操縦するには、空中でホイールを回すよりもはるかに小さいステアリング角が必要です。車輪の実際のステアリング角を測定するには、車両をトラックの表面に置くことが重要です。

- b. [センターステアリング] で、スライダーを徐々に動かすか、左または右矢印を押して、前輪の少なくとも 1 つが同じ側の後輪と一直線になる位置に合わせます。[次へ] を選択します。

Calibration > Calibrate steering angle

Step 1
Set your vehicle on the ground

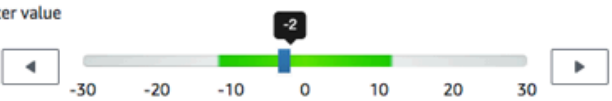
Step 2
Calibrate center

Calibrate steering angle

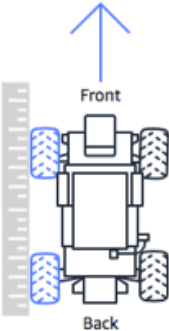
Center steering

Increase or decrease the **Center value** to center your vehicle. It is centered when any of the wheels points forward. Use a ruler or straight edge to ensure it is aligned with the rear wheel.

Center value



① The front wheels may not be perfectly aligned to each other -- it is important for one front wheel to be facing forward. DeepRacer uses Ackermann steering.



Cancel Previous Next

AWS DeepRacer は、[Ackermann の前輪ステアリング](#)を使用して、ターンの内側と外側で車輪を回転させます。これは一般的に左右の前輪が異なる角度で回転することを意味します。AWS DeepRacer では、キャリブレーションは中央値で行われます。したがって、選択した側の車輪を直線に揃えるように調整する必要があります。

Note

センターステアリングを可能な限りまっすぐに保つことができるように [AWS DeepRacer の車両を適切にキャリブレート](#) します。これをテストするには、手動で車両を押してまっすぐ進むことを確認します。

- c. [最大左ステアリング] で、車両の前輪がそれ以上左に動かなくなるまで、スライダーを徐々に左に動かすか、左矢印を押します。静かなノイズが聞こえます。大きな音が聞こえる場合は、行き過ぎています。この位置が最大左ステアリング角度に対応しています。シミュレートされたアクションスペースでステアリング角を制限している場合は、対応する値をここで一致させてください。[次へ] を選択します。

Calibration > Calibrate steering angle

Step 1
Set your vehicle on the ground

Step 2
Calibrate center

Step 3
Calibrate maximum left steering

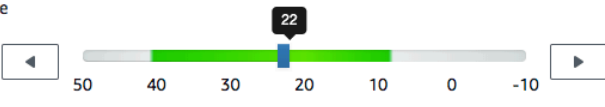
Step 4
Calibrate maximum right steering

Calibrate steering angle

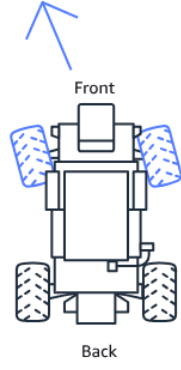
Maximum left steering

Increase the Value to turn the front wheels to the left until they stop turning.

Value



Estimated angle: 26-32°

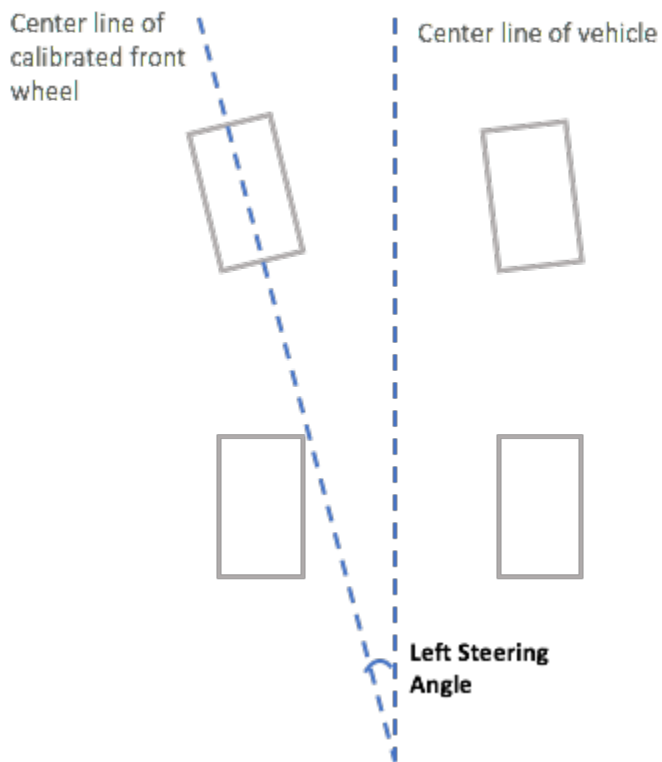


Front

Back

Cancel Previous Next

実際の最大左ステアリング角を測定するには、車両の中心線を引き、キャリブレーションのために選択した前輪の2つの端点をマークし、それが車両の中心線を横切るまでこの前輪の中心線を引きます。分度器を使用して角度を測定します。下の図を参照してください。トレーニングで実際の角度と一致させる場合は、次回のトレーニングジョブでアクションスペースに同じ値を設定できます。



- d. [最大右ステアリング] 下で、選択された車両の前輪がそれ以上右に動かなくなるまで、スライダを徐々に右に動かします。静かなノイズが聞こえます。大きな音が聞こえる場合は、行き過ぎています。この位置が最大右ステアリング角に対応しています。シミュレートされたアクションスペースでステアリング角を制限している場合は、対応する値をここで一致させてください。[完了] をクリックします。

Calibration > Calibrate steering angle

Step 1
Set your vehicle on the ground

Step 2
Calibrate center

Step 3
Calibrate maximum left steering

Step 4
Calibrate maximum right steering

Calibrate steering angle

Maximum right steering

Decrease the Value to turn the front wheels to the right until they stop turning.

Value

10 0 -10 -20 -30 -40 -50

Estimated angle: 26-32°

Front

Back

Cancel Previous Done

実際の最大右ステアリング角を測定するには、最大左ステアリング角の測定に使用したステップと同様のステップに従います。

これで AWS DeepRacer の車両のためのステアリングキャリブレーションは終了です。

4. 車両の最大スピードをキャリブレートするために、[キャリブレーション] ページ上の [スピード] 内にある [キャリブレートする] を選択し、以下のステップへ続きます。
 - a. 車輪が自由に回るように車両を持ち上げます。デバイスコントロールコンソールで [次へ] を選択します。

Calibration > Calibrate speed

Step 1
Raise your vehicle

Step 2
Calibrate stopped speed

Step 3
Set forward direction

Step 4
Calibrate maximum forward speed

Step 5
Calibrate maximum backward speed

Calibrate speed

Raise vehicle

Raise your vehicle to keep wheels from touching the ground and to key them moving freely.

Wheels spin at high speeds
Raise your vehicle on a stable surface when calibrating speed

Back Front

Cancel Next

Note

車両の速度の設定が高すぎると、キャリブレーション中に速度が速くなりすぎて、環境、車両、または近くの他のものに損傷を与える可能性があります。ここで指示されているように、車両を持ち上げる必要がありますが、手では持たないでください。

- b. 停止速度をキャリブレートするには、ホイールが回転しなくなるまで、デバイスコントロールコンソール上で [左矢印] または [右矢印] を押し、[停止スピード] の下で徐々に [停止値] を変更してください。[次へ] を選択します。

Calibration > Calibrate speed

Step 1
Raise your vehicle

Step 2
Calibrate stopped speed

Step 3
Set forward direction

Step 4
Calibrate maximum forward speed

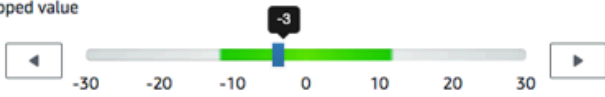
Step 5
Calibrate maximum backward speed

Calibrate speed

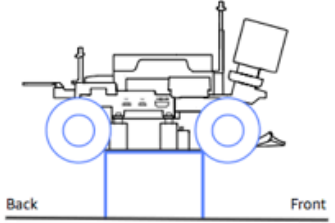
Stopped speed

With the vehicle's wheels free to spin, increase or decrease the **Stopped value** below until the wheels stop spinning.

Stopped value



Optimal range -20 through 20



Back Front

Cancel Previous Next

Note

ノイズが聞こえるようになったときに [停止値] をさらに左、またはさらに右に押し、車輪が動き出します。理想的なゼロスロットルポイントは、2つの値の間です。たとえば、左側の 16 と右側の -4 でノイズが聞こえ始める場合、最適な停止値は 10 になります。

- c. 車両の前進方向を設定するには、画面とイメージに表示されているように車両を置き、左右の矢印を押して車輪を回転させます。車輪が時計回りに回転すると、前進方向が設定されます。そうでない場合は、[逆方向] に切り替えます。[次へ] を選択します。

Calibration > Calibrate speed

Step 1
Raise your vehicle

Step 2
Calibrate stopped speed

Step 3
Set forward direction

Step 4
Calibrate maximum forward speed

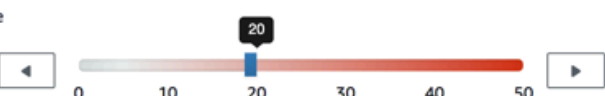
Step 5
Calibrate maximum backward speed

Calibrate Speed

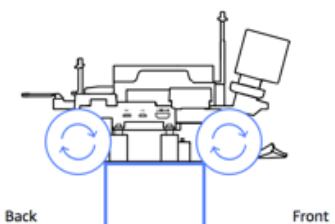
Set forward direction

Point the vehicle's front to the right as shown in the diagram. Push the left or right arrow to make the wheels turn. The vehicle will drive forward if the wheels turn clockwise.

Value



0 10 20 30 40 50



Back Front

⚠ If the wheels turn counter clock-wise, toggle on **Reverse direction**.

Reverse direction

Cancel Previous Next

Note

AWS re:Invent 2018 で配布される車両では、前方方向が逆に設定される場合があります。このような場合は、[逆方向] に切り替えてください。

- d. 最大前進速度をキャリブレートするには、[最大前進速度] で、スライダーをゆっくり左右に動かして、[最大前進速度の値] を徐々に正の値に調整し、[推定速度] の値が、シミュレーションで指定した最大速度と等しくなるようにします。[次へ] を選択します。

Calibration > Calibrate speed

Step 1
Raise your vehicle

Step 2
Calibrate stopped speed

Step 3
Set forward direction

Step 4
Calibrate maximum forward speed

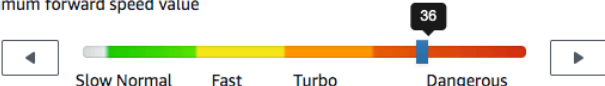
Step 5
Calibrate maximum backward speed

Calibrate speed

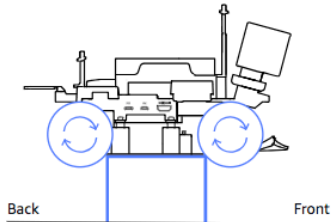
Maximum forward speed

Move the slider to set the maximum forward speed on the vehicle so that the **Estimated speed** value matches, precisely or approximately, the value specified in training the model that is or will be loaded to the vehicle's inference engine.

Maximum forward speed value



Estimated speed:
1.6 - 2.1 meters/second



Cancel Previous Next

Note

実際の最高速度は、路面の摩擦と車両のバッテリーレベルによって異なります。それを柔軟にするために、シミュレーションでトレーニング用に指定した最高速度より 20 ~ 30 パーセント高いように車両のスロットル限界を設定することができます。一般的に言えば、緑色の領域内で最高速度値を設定する必要があります。それ以上では、あまりにも速く走行する可能性があり、車が壊れる危険性が高まります。さらに、トレーニング用のアクションスペースは 2 m/s を超える最高速度をサポートしていません。

- e. 最大後退速度をキャリブレートするには、[最大後退速度] で、スライダーをゆっくり左右に動かして [最大後退速度の値] を徐々に負の値に調整し、[推定速度] がシミュレーションで指定した最大速度と等しくなるようにします。[完了] をクリックします。

Calibration > Calibrate speed

Step 1
Raise your vehicle

Step 2
Calibrate stopped speed

Step 3
Set forward direction

Step 4
Calibrate maximum forward speed


Step 5
Calibrate maximum backward speed

Calibrate speed

Maximum backward speed

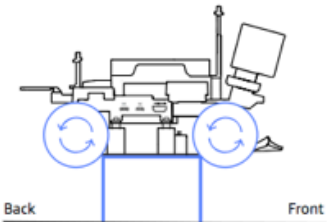
Move the slider to set the maximum backward speed on the vehicle so that the **Estimated speed** value matches, precisely or approximately, the value specified in training the model that is or will be loaded to the vehicle's inference engine.

Maximum backward speed value



Estimated speed

1.6 - 2.1 meters/second



Back Front

Cancel Previous Done

Note

自動運転モードでは、AWS DeepRacer 車両は後退速度を使用しません。後退速度は、車の手動運転モードを快適に制御できる任意の値に設定できます。

これでお客様の AWS DeepRacer 車両の最高速度のキャリブレーションは終了です。

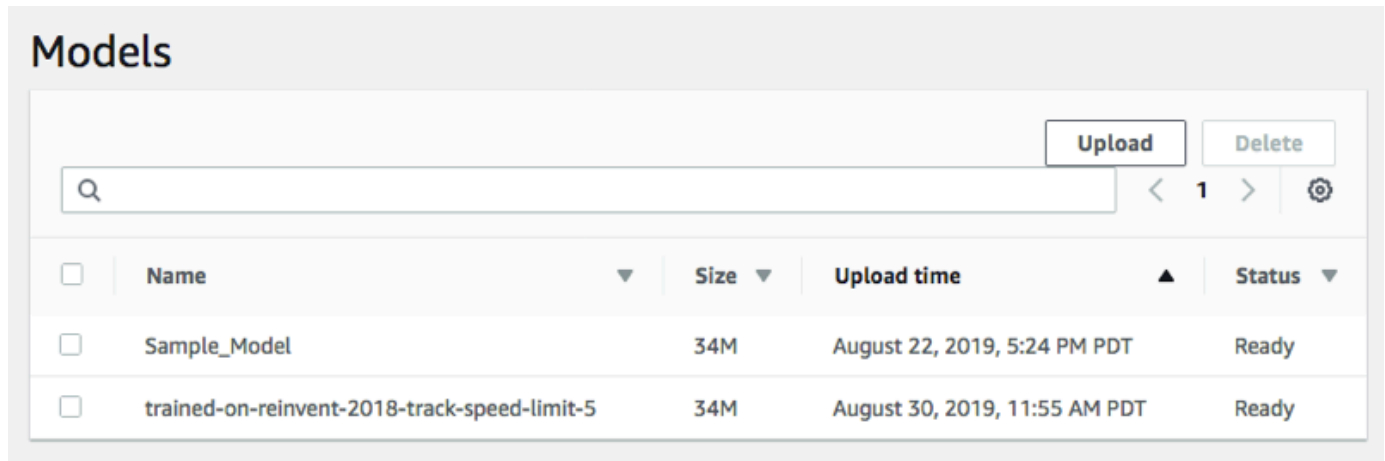
AWS DeepRacer 車両にモデルをアップロードする

AWS DeepRacer 車両を自動運転で開始するには、少なくとも 1 つの AWS DeepRacer モデルを AWS DeepRacer 車両にアップロードする必要があります。

モデルをアップロードするには、[モデルをトレーニングおよび評価](#)している必要があります。AWS DeepRacer コンソールを使用してモデルをトレーニングできます。その後、モデルアーティファクトをその Amazon S3 ストレージから、コンピュータからアクセス可能な（ローカルまたはネットワーク）ドライブにダウンロードする必要があります。

車両にトレーニングされたモデルをアップロードするには

1. デバイスコンソールのメインナビゲーションペインで、[モデル] を選択します。



2. 「モデル」ページで [モデル] のリストの上にある [アップロード] を選択します。
3. ファイルピッカーから、モデルアーティファクトをダウンロードしたドライブまたは共有に移動し、アップロードする圧縮モデルファイル (拡張子は *.tar.gz) を選択します。

正常にアップロードされたモデルのみが [モデル] リストに追加され、自動運転モードで車両の推論エンジンにロードできます。車両の推論エンジンにモデルをロードする方法については、「[AWS DeepRacer 車両を自律運転する](#)」を参照してください。

AWS DeepRacer 車両を運転する

[AWS DeepRacer 車両を設定](#)した後、車両の手動走行を始めるか、車両のデバイスコンソールを使用して自動走行させることができます。

自動運転の場合、AWS DeepRacerモデルをトレーニングし、トレーニングしたモデルアーティファクトを車両にデプロイしておく必要があります。自動レーシングモードでは、推論エンジンで走行しているモデルは、車両の方向と速度を制御します。トレーニング済みのモデルを車両にダウンロードしていない状態では、車両のデバイスコンソールを使用して、手動で車両を走行させることができます。

自動運転での車両のパフォーマンスは多数の要因に左右されます。要因には、トレーニングされたモデル、車両キャリブレーション、表面の摩擦、色のコントラストや照明の反射などのトラックの状態などがあります。車両が最適なパフォーマンスを実現するため、シミュレーションから実際へのモデルの移行が、正確で関連性があり有意義であることを確認する必要があります。詳細については、「[the section called “実環境のトレーニングを最適化する”](#)」を参照してください。

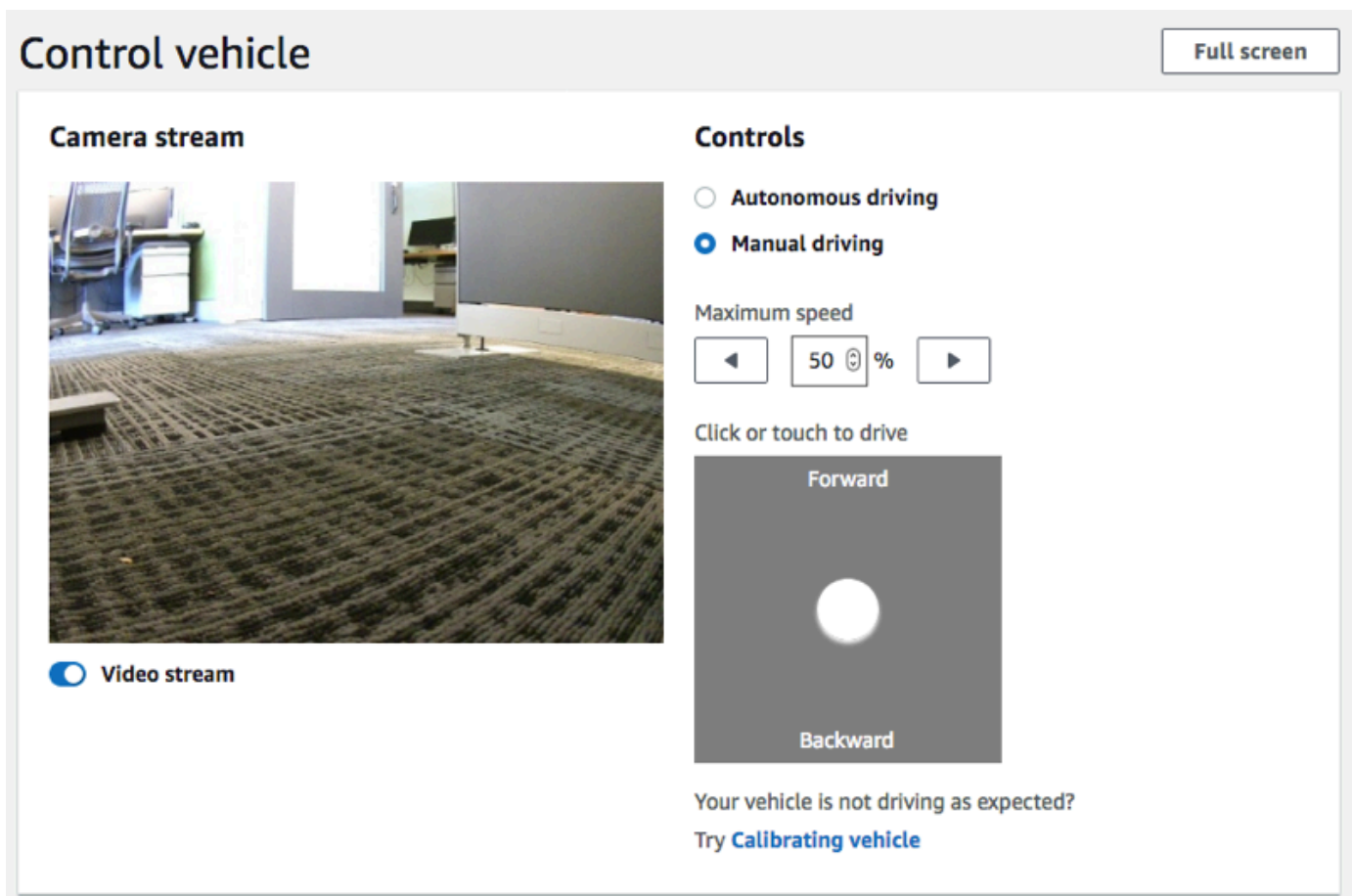
AWS DeepRacer 車両を手動で運転する

トレーニングをしたモデルがない場合、またはトレーニング済みのモデルの AWS DeepRacer 車両にデプロイしていない場合、自律動走行させることはできません。ただし、手動で走行させることはできます。

手動で AWS DeepRacer 車両を走行させるには、以下の手順に従ってください。

AWS DeepRacer 車両を手動走行させるには

1. AWS DeepRacer 車両を Wi-Fi ネットワークに接続した状態で、[指示](#)に従って、車両のデバイス管理コンソールにサインインします。
2. リポジトリの [制御車両] ページで、[コントロール] の下の [手動運転] を選択します。



3. [クリックまたはタッチして運転] で、運転パッド内の位置をクリックまたはタッチして車両を運転します。車両のフロントカメラからキャプチャされたイメージが、[カメラストリーム] のビデオプレーヤーに表示されます。
4. 運転中にデバイスコンソールでビデオストリームのオン・オフを切り替えるには、[カメラストリーム] 表示の下にある [ビデオストリーム] オプションのトグルを使用します。

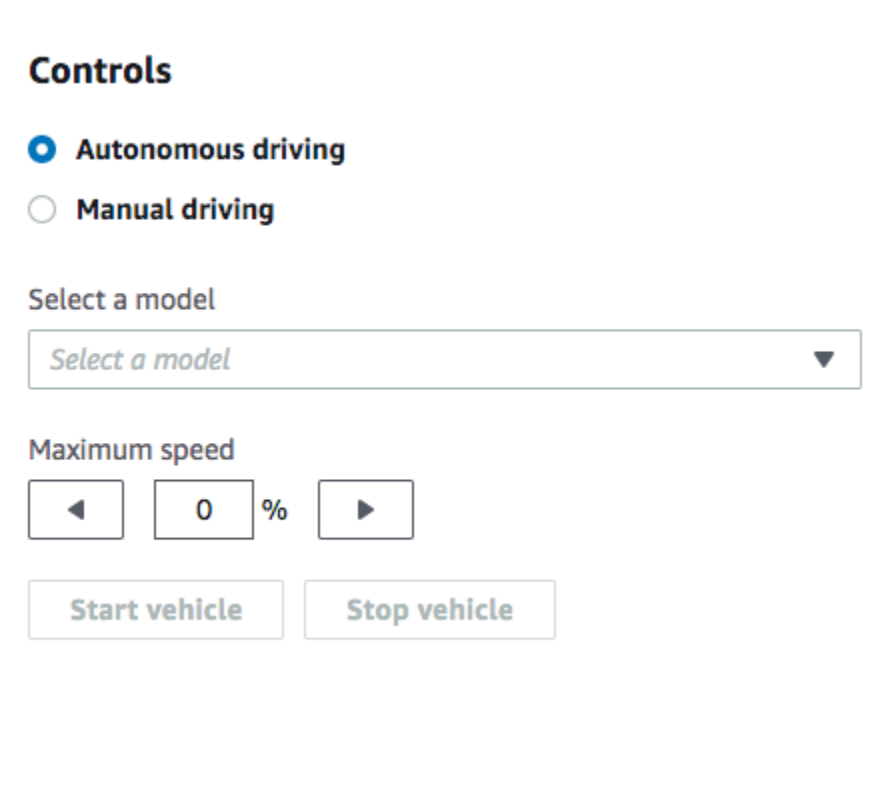
5. 車両を別の場所に移動するには、ステップ 3 以降を繰り返します。

AWS DeepRacer 車両を自律運転する

自律運転を開始するには、車両を物理的なトラックに配置し、次の操作を行います。

AWS DeepRacer 車両を自動走行させるには

1. [指示](#)に従って車両のデバイスコンソールにサインインし、自律運転のために以下の操作を実行します。
2. 「制御車両」ページで、[コントロール] の下にある [自律運転] を選択します。



Controls

Autonomous driving

Manual driving

Select a model

Select a model ▼

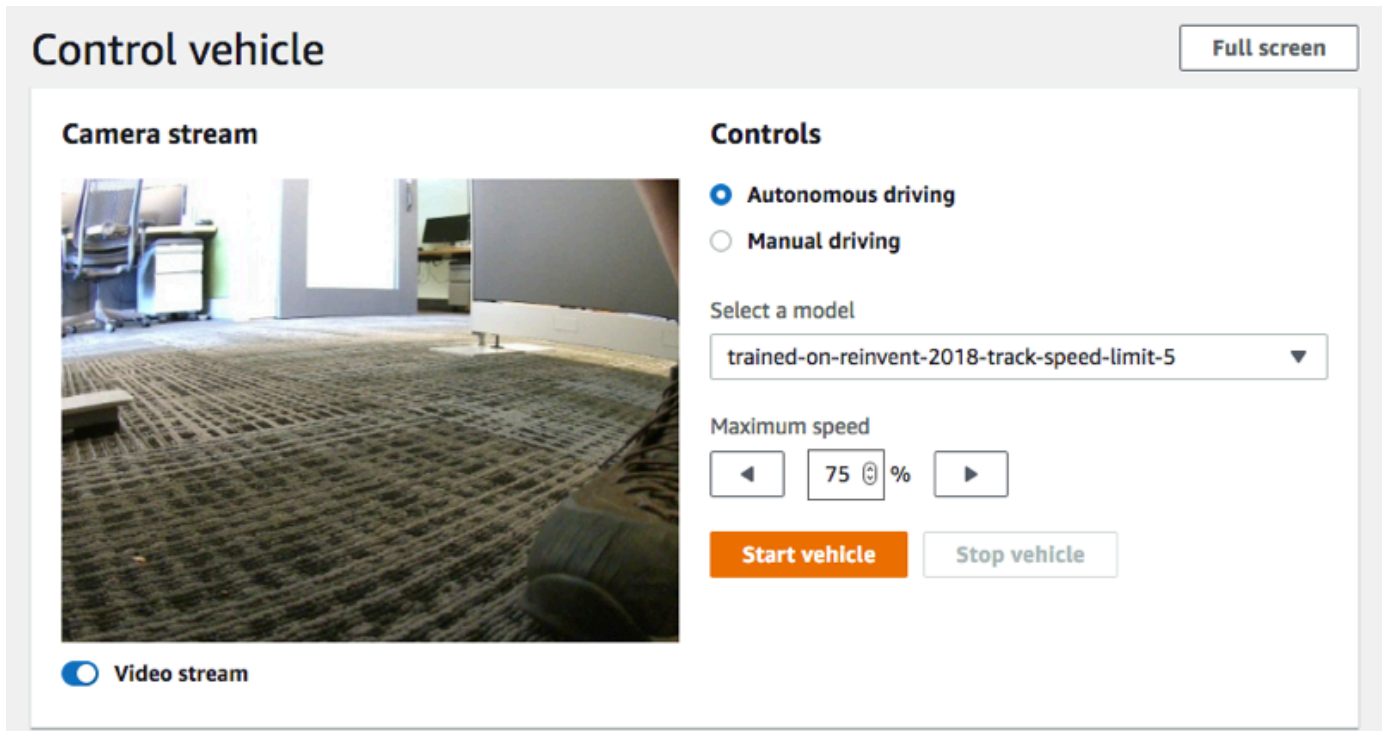
Maximum speed

◀ 0 % ▶

Start vehicle Stop vehicle

3. [モデルを選択] ドロップダウンリストから、アップロードされたモデルを選択します。続いて、[モデルをロードする] を選択します。これにより、モデルの推論エンジンへのロードが開始されます。このプロセスは約 10 秒で完了します。
4. 車両の [最大速度] 設定が、モデルのトレーニングに使用した最大速度の割合になるように調整します。

実際のトラックの表面摩擦などの要因によって、車両の最大速度がトレーニングで使用した最大速度より低下する可能性があります。最適な設定を試してみる必要があります。



5. [車両スタート] を選択して、車両を自律運転に設定します。
6. 運転中にデバイスコンソールでビデオストリームのオン・オフを切り替えるには、[カメラストリーム] 表示の下にある [ビデオストリーム] オプションのトグルを使用します。
7. 物理的なトラックまたはデバイスコンソールのストリーミングビデオのプレイヤーで車両が走るのを見ます。
8. 車両を停止するには、[車両を停止] を選択します。

同じモデルまたは異なるモデルで走行させるには、ステップ 3 から繰り返します。

AWS DeepRacer 車両の設定を確認、管理する

初回設定の後、デバイス管理コンソールを使用して、AWS DeepRacer 車両の設定を管理することができます。タスクには次のものが含まれています。

- 別の Wi-Fi ネットワークを選択する、
- デバイスコンソールパスワードをリセットする、
- デバイスの SSH 設定を有効または無効にする、
- 車両のトレイルライト LED の色を設定する
- デバイスソフトウェアおよびハードウェアバージョンを確認する、

- 車両のバッテリーレベルを確認する。

以下の手順では、これらのタスクについて説明します。

車両の設定を確認および管理するには

1. AWS DeepRacer 車両を Wi-Fi ネットワークに接続した状態で、[指示](#)に従って、車両のデバイス管理コンソールにサインインします。
2. メインのナビゲーションペインで、[設定] を選択します。
3. [設定] ページで、次の 1 つ以上のタスクを選択して実行します。

Settings

Network settings Edit

Wi-Fi network SSID	Vehicle IP address
--------------------	--------------------

Device console password Edit

Password *****

Device SSH Edit


SSH server Disabled	Password -
------------------------	---------------

LED color Edit

Color No color

About

AWS DeepRacer vehicle 1/18th scale 4WD monster truck chassis
Ubuntu OS 16.04.3 LTS, Intel® OpenVINO™ toolkit, ROS Kinetic

 Software up-to-date

Software version
Hardware version

Processor Intel Atom™ Processor
Memory 4GB RAM/Storage 32 GB memory (expandable)
Camera 4MP with MJPEG

- a. 別の Wi-Fi ネットワークを選択するには、[ネットワーク設定のための編集] を選択し、以下のステップに従います。
 - i. USB-to-USB-C ケーブルを使用して、車両をコンピュータに接続するには、[ネットワーク設定の編集] に表示される指示に従います。[USB 接続] ステータスが [接続済み] になった後、[deepracer.aws に移動する] ボタンを選択して、デバイスコンソールのログインページを開きます。

Settings > Edit network settings

Edit network settings


Network settings

Wi-Fi network SSID Mobile	IP address 10.92.206.61, 192.168.9.194	USB connection ⊗ Not connected
------------------------------	---	-----------------------------------

i Instructions

1. Connect your vehicle to your computer.

Use the included USB cable to connect your computer to the vehicle



- ii. デバイスコンソールのログインページで、車両の下部に印刷されているパスワードを入力し、[車両にアクセス] を選択します。
- iii. [Wi-Fi ネットワークの詳細] の下でドロップダウンリストから Wi-Fi ネットワークを選択し、選択したネットワークのパスワードを入力して、[接続] を選択します。

Wi-Fi network details

Specify your Wi-Fi network details.

Wi-Fi network name (SSID)

ATT807 ▼

Wi-Fi password

●●●●●●

Show password

Connect

- iv. Wi-Fi 接続の [車両ステータス] が [接続済み] になった後、[次へ] を選択してデバイスコンソールの [設定] ページに戻ります。このページに車両の新しい IP アドレスが表示されます。

- b. デバイスコンソールにサインインするためのパスワードをリセットするには、[デバイスコンソールパスワードのための編集] を選択し、以下のステップに従います。
 - i. [デバイスコンソールパスワードの編集] ページの [新しいパスワード] に新しいパスワードを入力します。
 - ii. パスワードの変更を確認するには、[パスワードを確認] にもう一度新しいパスワードを入力します。移動する前にパスワード値が同じである必要があります。
 - iii. タスクを完了するには、[パスワードの変更] を選択します。このオプションは、上記のステップで有効なパスワードの値を入力して確定した場合のみ、アクティブ化されません。

Settings > Edit device console password

Edit device console password

You are required to setup a password to protect access to your AWS DeepRacer vehicle. If you forget your password, [reset your password](#).

Old password

New password

Confirm password

Show passwords

[Change password](#)

- c. 車両への SSH 接続を有効または無効にするには、[SSH デバイスのための編集] を選択し、[有効化] または [無効化] を選択します。


Settings > Edit device SSH

Edit device SSH

SSH Server

Enable the SSH server on your device to enable login via CLI to execute command

Disabled Enabled

 Certain device functions such as software update are not supported over SSH.

4. 車両の LED の色を変更し、トラック上の車両を区別するには、「設定」ページの [LED の色のための編集] を開き、以下に従います。
 - a. 「LED の色を選択する」ページの [LED の色を選択する] というドロップダウンリストから利用可能な色を選択します。

Settings > Edit LED color

Edit LED color

Specify the behaviour and color of the LEDs located on the back of the vehicle.

Color

Select the color of the LEDs

Choose a color for the LEDs ▼

- Red
- Marigold
- Orchid Purple
- Sky Blue
- Green
- Violet
- Lime
- Default

Cancel

Save changes

同時にトラックを共有する他の車両から自分の車両を識別するのに役立つ色を選択する必要があります。

b. タスクを完了するには、[変更の保存] を選択します。

[変更の保存] 機能は、色を選択した後にのみアクティブになります。

5. デバイスのソフトウェアとハードウェアのバージョンを検査し、システムとカメラの設定を調べるには、[設定] の下の [バージョン情報] を確認します。
6. 車両のバッテリーの充電レベルを確認するには、プライマリナビゲーションペインの下部を確認します。

AWS DeepRacer 車両のログを表示する

AWS DeepRacer 車両は、車両の走行で発生する問題のトラブルシューティングに役立つ運用イベントをログに記録します。AWS DeepRacer 車両のログには 2 つのタイプがあります。

- システムイベントログは、プロセス管理、Wi-Fi 接続またはパスワードのリセットイベントなど車両のコンピュータオペレーティングシステムで発生しているオペレーションを追跡します。
- ロボットオペレーティングシステムは、車両の走行、ビデオストリーミング、およびポリシーの推論オペレーションを含む、ロボット操作に対して車両のオペレーティングシステムノードで発生している、操作のログレコードステータスをログに記録します。

デバイスログを表示するには、次のステップを実行します。

1. AWS DeepRacer 車両を Wi-Fi ネットワークに接続した状態で、[指示](#)に従って、車両のデバイス管理コンソールにサインインします。
2. コンソールのメインナビゲーションペインデバイスから [ログ] を選択します。
3. システムイベントを表示するには、[(イベントログイベンログ)] の下にスクロールします。

System event log



```

Apr 8 15:16:07 amss-42im login: message repeated 2 times: [ <INFO> Status returned from login proxy: 200]
Apr 8 15:16:07 amss-42im wifi_settings: <INFO> Check OTG Link State: not connected
Apr 8 15:16:07 amss-42im wifi_settings: <INFO> host: https://10.92.206.61/home otg_connected: not connected is_usb_connected: not
connected
Apr 8 15:16:07 amss-42im login: <INFO> Status returned from login proxy: 200
Apr 8 15:16:07 amss-42im login: message repeated 2 times: [ <INFO> Status returned from login proxy: 200]
Apr 8 15:16:07 amss-42im vehicle_control: <INFO> Changed the vehicle state to auto
Apr 8 15:16:07 amss-42im login: <INFO> Status returned from login proxy: 200
Apr 8 15:16:07 amss-42im wifi_settings: <INFO> Check OTG Link State: not connected
Apr 8 15:16:08 amss-42im utility: <INFO> Command executing: hostname -l
Apr 8 15:16:08 amss-42im utility: <INFO> ['10.92.206.61 192.168.9.194 ', '']
Apr 8 15:16:11 amss-42im login: <INFO> Status returned from login proxy: 200
Apr 8 15:16:41 amss-42im login: message repeated 3 times: [ <INFO> Status returned from login proxy: 200]
Apr 8 15:16:41 amss-42im ssh_api: <INFO> Providing ssh enabled as response
Apr 8 15:16:41 amss-42im utility: <INFO> Command executing: /bin/systemctl --no-pager status ssh
Apr 8 15:16:41 amss-42im wifi_settings: <INFO> Check OTG Link State: not connected
Apr 8 15:16:41 amss-42im utility: <INFO> * ssh.service - OpenBSD Secure Shell server#012 Loaded: loaded (/lib/systemd/system/ssh.service;
enabled; vendor preset: enabled)#012 Active: active (running) since Fri 2019-04-05 15:43:20 EDT; 2 days ago#012 Main PID: 16466 (sshd)#012
CGroup: /system.slice/ssh.service#012 └─16466 /usr/sbin/sshd -D#012#012Apr 08 14:37:07 amss-42im sshd[11396]: Accepted password for

```

4. ロボットオペレーティングシステムイベントを表示するには、[ロボットオペレーティングシステムログ]の下にあるイベントリストを下にスクロールします。

Robot operating system log



```

1554750920.064320544 Node Startup
1554750920.131309136 INFO [/opt/workspace/AwsSilverstoneDeviceLib/ros-src/servo_pkg/src/servo_node.cpp:439(LedMgr::LedMgr) [topics:
/rosout] LedMgr pwm channel creation
1554750920.201161384 INFO [/tmp/binarydeb/ros-kinetic-roscpp-1.12.14/src/libros/service.cpp:80(service::exists) [topics: /rosout]
waitForService: Service [/media_state] has not been advertised, waiting...
1554750920.640698003 INFO [/tmp/binarydeb/ros-kinetic-roscpp-1.12.14/src/libros/service.cpp:122(service::waitForService) [topics: /rosout]
waitForService: Service [/media_state] is now available.
1554750920.578106989 INFO [/opt/workspace/AwsSilverstoneDeviceLib/ros-src/web_video_server
/src/web_video_server.cpp:96(WebVideoServer::spin) [topics: /rosout] Waiting For connections on 0.0.0.0:8080
1554750921.752294063 INFO [navigation_node.py:154(set_action_space_scales) [topics: /auto_drive, /rosout, /rl_results] Action space scale set:
{'steering_max': 30.0, 'speed_max': 0.8}
Mapping equation params a: -1.875 b: 2.75
1554750930.167246103 INFO [software_update_process.py:25(logger) [topics: /rosout] /software_update: [04/08/19 15:15:30] Setup Ethernet
over OTG.
1554750930.174333095 INFO [software_update_process.py:25(logger) [topics: /rosout] /software_update: [04/08/19 15:15:30] Entering
daemon loop.
1554750930.205965042 INFO [software_update_process.py:25(logger) [topics: /rosout] /software_update: [04/08/19 15:15:30] Updating
network information.
1554750930.209075927 INFO [software_update_process.py:25(logger) [topics: /rosout] /software_update: [04/08/19 15:15:30] Checking
software update...
1554750938.287539958 INFO [software_update_process.py:25(logger) [topics: /rosout] /software_update: [04/08/19 15:15:38] Verifying
package aws-deepracer-core...

```

AWS DeepRacer デバイスを更新して復元する

AWS DeepRacer デバイスを Ubuntu 20.04 Focal Fossa、Intel® OpenVino™ ツールキット 2021.1.110、ROS2 Foxy Fitzroy、Python 3.8 を含む最新のソフトウェアスタックに更新して復元してください。この更新は AWS DeepRacer オープンソースプロジェクトを実行するために必要ですが、それ以外の場合はオプションです。AWS DeepRacer は Ubuntu 20.04 Focal Fossa と ROS2 Foxy Fitzroy のみをサポートしています。

Important

新しい AWS DeepRacer ソフトウェアスタックにアップデートすると、お使いの AWS DeepRacer デバイス上のすべてのデータが消去されます。

トピック

- [AWS DeepRacer デバイスが現在実行しているソフトウェアバージョンをチェックする](#)
- [AWS DeepRacer デバイスのソフトウェアを Ubuntu 20.04 スタックに更新する準備を整えるには](#)
- [AWS DeepRacer デバイスを Ubuntu 20.04 ソフトウェアスタックに更新します。](#)

AWS DeepRacer デバイスが現在実行しているソフトウェアバージョンをチェックする

AWS DeepRacer デバイスが現在実行しているソフトウェアバージョンをチェックする

1. AWS DeepRacer 車両コンソールにログインします。方法については、[the section called “デバイスコンソールを起動する”](#) のステップに従います。
2. ナビゲーションペインで [設定] を選択します。
3. [バージョン情報] セクションで、AWS DeepRacer Vehicle が現在実行しているソフトウェアバージョンをチェックします。

About

AWS DeepRacer vehicle 1/18th scale 4WD monster truck chassis
Ubuntu OS 20.04.1 LTS, Intel® OpenVINO™ toolkit, ROS2 Foxy

✔ Software up-to-date

Software version 2.0.113.0

Hardware version R2.1

Processor Intel Atom™ Processor

Memory 4GB RAM/Storage 32 GB memory (expandable)

Camera 4MP with MJPEG

AWS DeepRacer デバイスのソフトウェアを Ubuntu 20.04 スタックに更新する準備を整えるには

このトピックでは、AWS DeepRacer Ubuntu インストールメディアの作成手順について説明します。ブート可能な USB ドライブを準備するには、追加のハードウェアが必要です。

前提条件

開始する前に、以下の項目を用意したことを確認します。

- AWS DeepRacer デバイス
- 1 台の USB フラッシュドライブ (32 GB 以上)
- カスタム AWS DeepRacer [Ubuntu ISO イメージ](#)。
- 最新の AWS DeepRacer [ソフトウェア アップデートのパッケージ](#)。
- ご使用のオペレーティングシステムと互換性のある [UNetbootin](#) のコピー。
- Ubuntu、Windows、または macOS を実行して USB インストールメディアを準備しているコンピューター。HDMI タイプ A ケーブルでマウス、キーボード、モニターを接続することで、AWS DeepRacer デバイスのコンピューティング モジュールを Linux コンピューターとして使用することもできます。

準備

AWS DeepRacer アップデートメディアを準備するには、次の作業を実行します。

- USB ドライブを以下の 2 つのパーティションにフォーマットします：
 - 4 GB の FAT32 ブートパーティション
 - 18 GB 以上の NTFS データパーティション
- 再起動時にアップデートを開始するために USB ドライブをブート可能にします。
 - 必要なカスタム Ubuntu ISO イメージをブートパーティションに書き込みます。
 - アップデートに必要なファイルを USB ドライブのデータパーティションにコピーします。

ブート可能な USB ドライブを準備する

以下の手順に従って、Ubuntu (Linux)、Windows、または macOS で AWS DeepRacer アップデートメディアを準備します。使用するコンピュータにより、特定のタスクがオペレーティングシステムによって異なる場合があります。オペレーティングシステムに対応するタブを選択します。

Ubuntu

こちらの手順に従って、AWS DeepRacer デバイスのコンピューティングモジュールを含む Ubuntu コンピュータを使用して、AWS DeepRacer デバイス用のアップデートメディアを準備します。別の Linux ディストリビューションを使用している場合は、`apt-get * コマンド`をオペレーティングシステムのパッケージマネージャと互換性のあるものに置き換えます。

USB ドライブを消去してパーティション分割するには

1. 次のコマンドを実行して GParted をインストールして起動します。

```
sudo apt-get update; sudo apt-get install gparted
sudo gparted
```

2. USB ドライブを消去するには、そのデバイスパスが必要です。GParted コンソールで検索して USB ドライブを消去するには、次の操作を行います。
 - a. メニューバーで [表示] を選択し、[デバイス情報] を選択します。選択したディスクのモデル、サイズ、パスを示すサイドバーが表示されます。
 - b. メニューバーで GParted を選択してから、[デバイス] に移動して USB ドライブを選択し、最後にリストから USB ドライブを選択します。[デバイスの説明] に表示されているサイズとモデルを USB ドライブと一致させてください。
 - c. 正しいディスクが選択されていることを確認した後、既存のパーティションをすべて削除します。

パーティションがロックされている場合、コンテキスト (右クリック) メニューを開き、[アンマウント] を選択します。

3. 4 GB 容量の FAT32 ブートパーティションを作成するには、左上にあるファイルアイコンを選択し、次に示すパラメータを設定して、[追加] を選択します。

[現パーティションの前方にこの分空き容量を作る:] **1**

[現パーティションの変更後の容量:] **4096**

[現パーティションの後方にこの分空き容量を作る:] **<remaining size>**

[アラインメントの位置:] **MiB**

[作成するタイプ:] **Primary Partition**

[パーティション名:]

[ファイルシステム:] **fat32**

[ラベル:] **BOOT**

4. 18 GB 以上の容量の NTFS データパーティションを作成するには、もう一度ファイルアイコンを選択し、次に示すパラメータを設定し、[追加] を選択します。

[現パーティションの前方にこの分空き容量を作る:] **0**

[現パーティションの変更後の容量:] **<remaining size>**

[現パーティションの後方にこの分空き容量を作る:] **0**

[アラインメントの位置:] **MiB**

[作成するタイプ:] **Primary Partition**

[パーティション名:]

[ファイルシステム:] **ntfs**

[ラベル:] **Data**

5. メニューバーで [編集] を選択し、[すべての操作を適用] を選択します。変更を適用するかどうかを確認する警告プロンプトが表示されます。[適用] を選択します。

6. FAT32 パーティションと NTFS パーティションを作成すると、USB ドライブのパーティション情報が GParted コンソールに表示されます。BOOT パーティションのドライブパスを書きとめておきます。次のステップを完了するために必要になります。

FAT32 パーティションから USB ドライブをブート可能にするには

1. 前提条件セクションから、[カスタム Ubuntu ISO イメージ](#)をダウンロードしたことを確認してください。
2. Ubuntu 20.04 を使用している場合は、そのバイナリファイルを使用して UNetbootin を実行する必要があります。これを実行するには:
 - a. 最新の [UNetbootin バイナリファイル](#)をダウンロードフォルダにダウンロードします。この例では、`unetbootin-linux64-702.bin` を使用します。
 - b. `<Ctrl> + <Alt> + <T>` を押してターミナルを開きます。または、メニューバーの [アクティビティ] を選択し、検索バーに `terminal` を入力して [ターミナル] アイコンを選択します。
 - c. 次のコマンドを使用してバイナリファイルの場所に移動し、ファイルに実行権限を付与して UNetbootin を実行します。ダウンロードしたバイナリファイルのバージョンと一致しない場合は、コマンド内のファイル名を必ず調整してください。

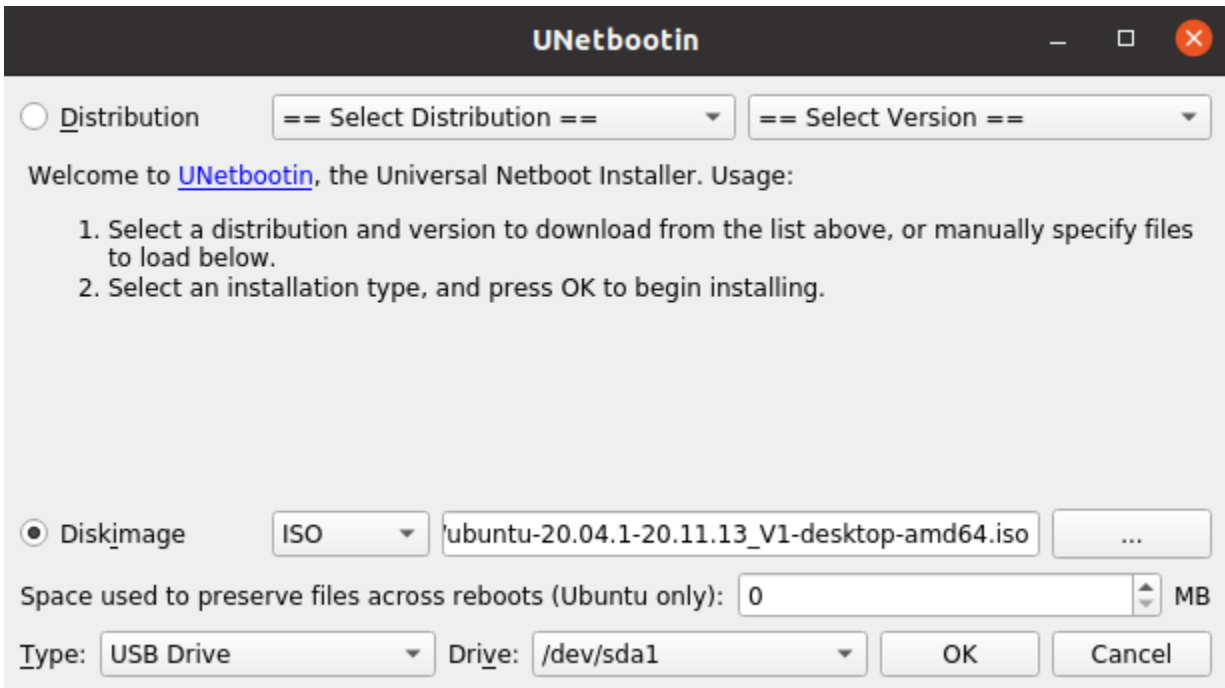
```
cd Downloads
sudo chmod +x ./unetbootin-linux64-702.bin
sudo ./unetbootin-linux64-702.bin
```

古いバージョンの Ubuntu を使用している場合は、以下のコマンドを実行して UNetbootin をリポジトリからインストールします。

```
sudo add-apt-repository ppa:gezakovacs/ppa
sudo apt-get update; sudo apt-get install unetbootin
sudo unetbootin
```

3. UNetbootin コンソールで、以下の操作を行います。
 - a. [ディスクイメージ] のラジオボタンをチェックします。
 - b. ディスクイメージのタイプでは、ドロップダウンメニューから、ISO を選択します。
 - c. ファイルセレクターを開き、前提条件セクションに記載されている [Ubuntu ISO](#) を選択します。

- d. [タイプ] では、[USB ドライブ] を選択します。
- e. [ドライブ] では、BOOT パーティションのドライブパスを選択します (例では「`/dev/sda1`」)。
- f. [OK] を選択してください。



i Tip

[/dev/sda1 not mounted] (/dev/sda1 がマウントされていません) という警告が表示された場合は、[OK] を選択して警告を閉じてから USB ドライブを取り外し、ドライブを再度挿入してから上記のステップを実行して、Ubuntu ISO イメージを作成します。

AWS DeepRacer アップデートファイルを NTFS パーティションに展開するには

1. 前提条件セクションからダウンロードした [ソフトウェアアップデートパッケージ](#) を解凍します。
2. アップデートパッケージの内容を USB ドライブのデータ (NTFS) パーティションのルートに展開します。

Windows

Windows コンピュータを使用して AWS DeepRacer デバイス用のアップデートメディアを準備するには、こちらの手順に従います。

USB ドライブを消去するには

1. Windows のコマンドプロンプトを開き、`diskpart` を入力し、[OK] を選択して、Windows の DiskPart コンソールを起動します。
2. Microsoft DiskPart のターミナルが開いたら、`list disk` を [DISKPART>] の後に入力し、空にする USB ドライブを検索する為に使用可能なディスク一覧を表示します。
3. お客様の USB ドライブと一致するディスクを選択してください。例えば、`select Disk 2` を [DISKPART>] の後に入力します。次のステップは復元が出来ないため、表示画面を注意深く読み、空にしたいディスクが選択されていることを確認します。
4. 正しいディスクが選択されていることを確認した後、`Clean` を [DISKPART>] プロンプトの後に入力します。
5. `list disk` を [DISKPART>] の後に再度入力します。空にしたディスクを探し、容量と残量を照らし合わせます。2 つの値が一致すれば、成功となります。
6. Windows DiskPart コンソールを終了するには、`Exit` を [DISKPART>] プロンプトの後に入力します。

USB ドライブをパーティション分割するには

1. Windows コマンドプロンプトを開き、`diskmgmt.msc` と入力した後に [OK] を選択して、Windows の ディスク管理コンソールを起動します。
2. ディスク管理コンソールから、USB ドライブを選択します。
3. 4 GB の容量の FAT32 パーティションを作成するには、USB ドライブの未割り当て領域でコンテキスト (右クリック) メニューを開き、[新規シンプルボリューム] を選択します。[新規シンプルボリューム] ウィザードが表示されます。
4. [新規シンプルボリューム] ウィザードが表示されたら、次の操作を実行します。
 - a. [ボリュームサイズの指定] ページで、次のパラメータを設定し、[次へ] を選択します。

シンプルボリュームサイズ (MB 単位): **4096**
 - b. [ドライブ文字またはパスの割り当て] ページで、[次のドライブ文字を割り当てる:] ラジオボタンをオンにし、ドロップダウンリストからドライブ文字を選択して、[次へ] を選

択します。割り当てたドライブ文字を書き留めておきます。後で FAT32 パーティションを起動可能にするときに必要になります。

- c. [パーティションをフォーマット] ページで、[このボリュームを次の設定でフォーマットする] ラジオボタンをオンにし、次のパラメータを設定して、[次へ] を選択します。

ファイルシステム: **FAT32**

割り当て単位のサイズ: **Default**

ボリュームラベル: **BOOT**

[クイックフォーマットを実行] にチェックを入れたままにします。

5. 残りのディスク容量で NTFS パーティションを作成するには、USB ドライブの残りの未割り当て領域でコンテキスト (右クリック) メニューを開き、[新規シンプルボリューム] を選択します。[新規シンプルボリューム] ウィザードが表示されます。
6. [新規シンプルボリューム] ウィザードが表示されたら、次の操作を実行します。
 - a. [ボリュームサイズの指定] ページで、[シンプルボリュームサイズ (MB)] を [最大ディスク容量 (MB)] と一致するように設定し、[次へ] を選択します。
 - b. [ドライブ文字またはパスの割り当て] ページで、[次のドライブ文字を割り当てる:] ラジオボタンをオンにし、ドロップダウンリストからドライブ文字を選択して、[次へ] を選択します。
 - c. [パーティションをフォーマット] ページで、[このボリュームを次の設定でフォーマットする] ラジオボタンをオンにし、次のパラメータを設定して、[次へ] を選択します。

ファイルシステム: **NTFS**

割り当て単位のサイズ: **Default**

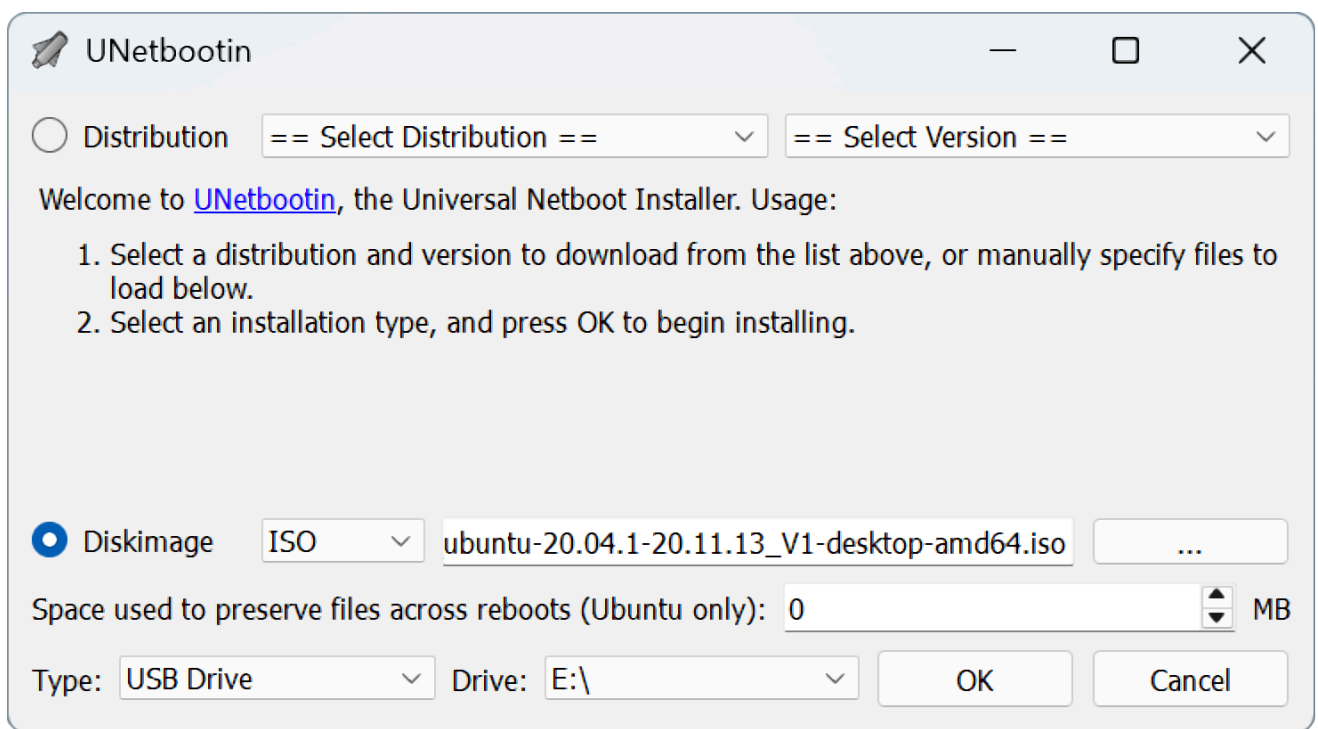
ボリュームラベル: **Data**

[クイックフォーマットを実行] にチェックを入れたままにします。

FAT32 パーティションから USB ドライブをブート可能にするには

1. 前提条件セクションから [カスタマイズした Ubuntu ISO イメージ](#) をダウンロードしたことを確認してください。
2. [UNetbootin](#) をダウンロードしたら、UNetbootin コンソールを起動します。

3. UNetbootin コンソールで、以下の操作を行います。
 - a. [ディスクイメージ] のラジオボタンを確認します。
 - b. [ディスクイメージ] には、ドロップダウンメニューから ISO を選択します。
 - c. ファイルピッカーを開き、Ubuntu ISO ファイルを選択します。
 - d. [タイプ] では、[USB ドライブ] を選択します。
 - e. [ドライブ] では、作成した FAT32 パーティションに対応したドライブ文字を選択します。ここでは E:\ です。
 - f. [OK] を選択してください。



AWS DeepRacer アップデートファイルを NTFS パーティションに展開するには

1. 前提条件セクションからダウンロードした[ソフトウェアアップデートパッケージ](#)を解凍します。

i Tip

希望するツールでファイルを正常に解凍できない場合は、PowerShell の [Expand-Archive](#) コマンドを使用します。

2. アップデートパッケージの内容を USB ドライブのデータ (NTFS) パーティションのルートに展開します。

macOS

Mac を使用して AWS DeepRacer デバイス用のアップデートメディアを準備するには、こちらの手順に従います。

USB ドライブを消去してパーティション分割するには

1. Mac に USB ドライブを接続します。
2. Command + Space バーを押して [スポットライト] 検索フィールドを開き、Disk Utility と入力します。

または、[ファインダー] -> [アプリケーション] -> [ユーティリティ] -> [ディスクユーティリティ] の順に選択して、ディスクユーティリティを開きます。

3. メニューバーで [表示] を選択し、[すべてのデバイスを表示] を選択します。
4. サイドバーの [外付け] で、フォーマットする USB ドライブを選択し、[消去] を選択します。
5. 新しいウィンドウが開き、USB ドライブを消去するかどうかの確認が求められ、[名前]、[フォーマット]、および [パーティションスキーム] を変更できます。まだ名前を変更する必要はありません。[フォーマット] と [スキーム] で次のオプションを選択し、[消去] を選択します。

- フォーマット: Mac OS 拡張 (ジャーナル機能付き)
- スキーム: GUID パーティションマップ

消去処理が完了したら、ダイアログウィンドウで [完了] を選択します。

6. ディスクユーティリティのメインウィンドウで、サイドバーから USB ドライブを選択し、上部のツールバーから [パーティション] を選択します。[パーティションデバイス####USB##

##ですか?]というタイトルのポップアップウィンドウが表示されます。[追加 (+)] ボタンを選択して、新しいパーティションを作成します。

7. 新しいパーティションを作成したら、[パーティション情報] で、次の項目を選択して入力します。

- 名前: **BOOT**
- フォーマット: MS-DOS (FAT)
- サイズ: 4 GB

 Tip

フォーマットとして MS-DOS (FAT) を選択した後に [サイズ] の入力ボックスがグレー表示になっている場合は、**BOOT** パーティションが 4 GB になるまでパーティショングラフのサイズ変更コントロールをドラッグできます。

まだ [適用] を選択しないでください。

8. もう 1 つの [無題] パーティションを選択し、[パーティション情報] で次のオプションを選択して入力します。

- 名前: **Data**
- フォーマット: ExFAT
- サイズ: USB ドライブの残りの容量 (GB 単位)

[Apply] (適用) を選択します。

9. 新しいウィンドウが開き、USB ドライブに加えられる変更が表示されます。これらの変更が正しいことを確認してください。確認して新しいパーティションの作成を開始するには、[パーティション] を選択します。
10. Disk Utility コンソールで、サイドバーから **BOOT** パーティションを選択し、ツールバーから情報を選択します。BSD デバイスノードの値を書き留めておいてください。このチュートリアルで使用した値とは異なる場合があります。この場合、割り当てられる値は `disk4s2` です。FAT32 パーティションから USB ドライブをブート可能にするには、このパスを指定する必要があります。

FAT32 パーティションから USB ドライブをブート可能にするには

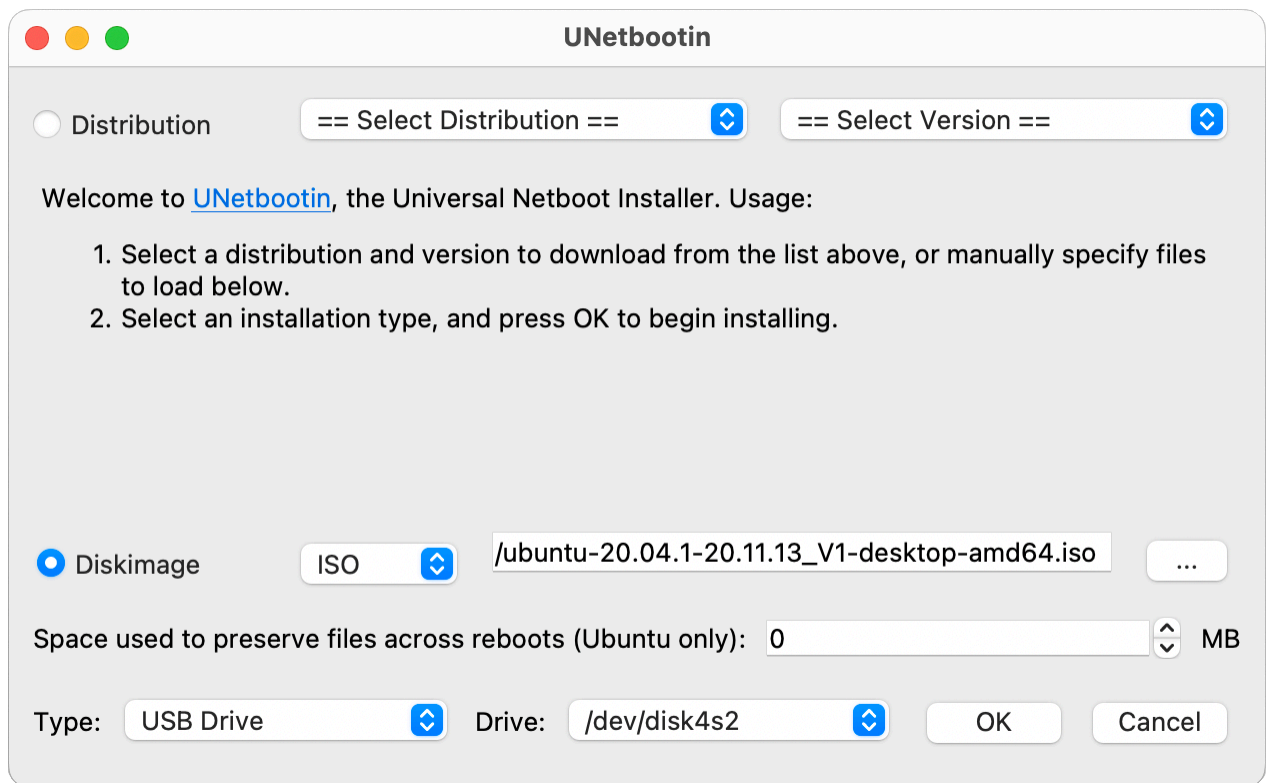
1. 前提条件セクションから [カスタマイズした Ubuntu ISO イメージ](#) をダウンロードしたことを確認してください。
2. [UNetbootin](#) をダウンロードした後、コンテキスト (右クリック) メニューから [開く] を選択します。アプリケーションを開くかどうかを確認するセキュリティプロンプトが表示されるので、[開く] を選択して UNetbootin コンソールを起動します。

[Apple Silicon を搭載した Mac](#) を使用していて、[開く] を選択しても UNetbootin コンソールが表示されない場合は、次の手順に従って Rosetta 2 がインストールされていることを確認してください。

- a. [ファインダー] > [アプリケーション] > [ユーティリティ] > [ターミナル] を選択してターミナルウィンドウを開きます。
- b. 次のコマンドを入力して、Rosetta 2 をインストールします:

```
softwareupdate --install-rosetta
```

- c. UNetbootin をもう一度開いてください。
3. UNetbootin コンソールで、以下の操作を行います。
 - a. [ディスクイメージ] のラジオボタンを確認します。
 - b. [ディスクイメージ] には、ドロップダウンメニューから ISO を選択します。
 - c. ファイルピッカーを開き、Ubuntu ISO ファイルを選択します。
 - d. [タイプ] では、[USB ドライブ] を選択します。
 - e. [ドライブ] では、BOOT パーティション用の BSD デバイスノード (この例では /dev/disk4s2) を選択します。
 - f. [OK] を選択してください。



i Tip

[/dev/disk2s2 がマウントされていません] という警告が表示された場合は、[OK] を選択してメッセージを閉じ、USB ドライブを取り外し、ドライブを再接続してから、上記のステップを実行して、Ubuntu ISO イメージを作成します。

AWS DeepRacer アップデートファイルを ExFAT パーティションに展開するには

1. 前提条件セクションからダウンロードした [ソフトウェアアップデートパッケージ](#) を解凍します。
2. アップデートパッケージの内容を USB ドライブのデータ (ExFAT) パーティションのルートに展開します。

AWS DeepRacer デバイスを Ubuntu 20.04 ソフトウェアスタックに更新します。

前のステップで説明したように USB アップデートメディアを作成すると、Ubuntu 20.04 Focal Fossa、Intel® OpenVino™ ツールキット 2021.1.110、ROS2 Foxy Fitzroy、Python 3.8 を含む、最新のソフトウェアスタックに AWS DeepRacer デバイスをアップデートできます。

Important

新しい AWS DeepRacer ソフトウェアスタックにアップデートすると、お使いの AWS DeepRacer デバイス上のすべてのデータが消去されます。

AWS DeepRacer デバイスのソフトウェアを Ubuntu 20.04 スタックにアップデートするには

1. AWS DeepRacer 車両をモニタリングに接続します。HDMI-HDMI ケーブル、HDMI-DVI ケーブル、または同様のケーブルが必要です。ケーブルの HDMI 端をコンピューティングモジュールの HDMI ポートに挿入し、もう一方の端をモニター上の互換性のあるポートに接続します。
2. USB キーボードとマウスを接続します。車両の前面と両側には、カメラを接続可能なポートを含め、3 つの AWS DeepRacer コンピューティング モジュールの USB ポートがあります。4 番目の USB ポートはコンピューティングバッテリーと LED テールライトの間のスペースにあります。
3. コンピューティングモジュールの空いている USB ポートに、USB アップデートメディアを挿入します。電源を入れるか、AWS DeepRacer デバイスをリセットし、<ESC> キーを繰り返し押し続けて BIOS に移行します。
4. BIOS ウィンドウから [ファイルから起動] を選択し、ブートパーティションの名前 (この例では BOOT という名前) でオプションを選択します。次に、<EFI> を選択してから、<BOOT> を選択し、最後に BootX64.EFI を選択します。
5. コンピューティングモジュールが起動すると、デスクトップにターミナルウィンドウが表示され、進行状況が表示されます。AWS DeepRacer デバイスは 10 秒後に自動的に更新プロセスを開始します。この段階では、これ以上のユーザー入力はありません。

エラーが発生して、更新が失敗した場合、ステップ 1 から手順を再開します。エラーメッセージの詳細については、USB ドライブのデータパーティションで生成された `result.log` ファイルを参照してください。

6. ソフトウェアのアップデートが完了するまで待ちます。工場出荷時設定へのリセットが完了すると、ターミナルウィンドウは自動的に閉じます。
7. デバイスソフトウェアのアップデートが完了した後は、コンピューティング モジュールから USB ドライブを切り離します。これで、AWS DeepRacer デバイスを再起動またはシャットダウンできます。
8. 更新後、AWS DeepRacer デバイスのユーザー認証情報は、デフォルトで以下の値になります。初回ログイン時にパスワードを変更するよう求められます。

ユーザー: Deepracer

パスワード: deepracer

AWS DeepRacer の物理的なトラックを構築する

このセクションでは、AWS DeepRacer モデルの物理的なトラックを構築する方法について説明します。AWS DeepRacer を自動走行させ、強化学習モデルを物理的環境でテストするには、物理的なトラックが必要です。トラックはトレーニングで使用されたシミュレートされたトラックに似ていて、デプロイされた AWS DeepRacer モデルをトレーニングするのに使用された環境を複製したものです。

最適なエクスペリエンスを得るには、事前にプリントされたトラックとトラック障壁を使用するようお勧めします。事前にプリントされたトラックと障壁を使用すると、AWS DeepRacer トラック環境のセットアップと設置が容易になります。トラックを最初から構築するのではなく、あらかじめプリントされたトラックとトラック障壁を組み立てます。イベント終了後は、分解して保管し、プリント済みのトラックや障壁を今後のイベントで再利用することができます。事前にプリントされたトラックと障壁、およびイベントのスペースや、その他の要件の見積もりに関する詳細は、「[AWS DeepRacer ストアフロント](#)」で入手できます。

トピック

- [トラックの材料とビルドツール](#)
- [AWS DeepRacer トラックの配置](#)
- [AWS DeepRacer トラック設計テンプレート](#)

トラックの材料とビルドツール

トラックの構築を開始する前に、次の材料とツールを用意してください。

トピック

- [必要な材料](#)
- [必要なツール](#)

必要な材料

トラックを構築するには、次の材料が必要です。

- トラックの境界:

濃い色のトラック表面に対して約 2 インチ幅の白またはオフホワイト色のテープでトラックを作成できます。濃い色の表面に、白またはオフホワイトのテープを使用します。たとえば、[1.88 インチ幅の、パールホワイトのダクトテープ](#)または [1.88 インチ \(粘着性が少ない\) のマスキングテープ](#)。

- **トラックの表面:**

硬木、じゅうたん、コンクリート、または[アスファルトフェルト](#)などの濃い色の硬い床にトラックを作成できます。後者は、最小限の反射で実世界の路面に似せて作られています。[連動フォーム](#)または[ゴムパッド](#)も良い選択肢です。

必要なツール

次のツールは、トラックの設計および構築に必要なものであるか、または役に立ちます。

- **巻き尺およびはさみ**

適切な巻尺とはさみはトラックを構築するために不可欠です。お持ちでない場合は、[巻尺はこちら](#)で、[はさみはこちら](#)で注文できます。

- **オプションの設計ツール**

独自のトラックを設計するには、[分度器](#)、[定規](#)、[鉛筆](#)、[ナイフ](#)および[コンパス](#)が必要です。

AWS DeepRacer トラックの配置

トラックを構築する際は、直線またはシングルターンのトラックなどの単純なデザインから始めることをお勧めします。次に、ループしたトラックに進むことができます。ここでは、例としてシングルターントラックを使用して、独自のトラックを作成するステップについて説明します。まず、トラックの寸法要件を見てみましょう。

トピック

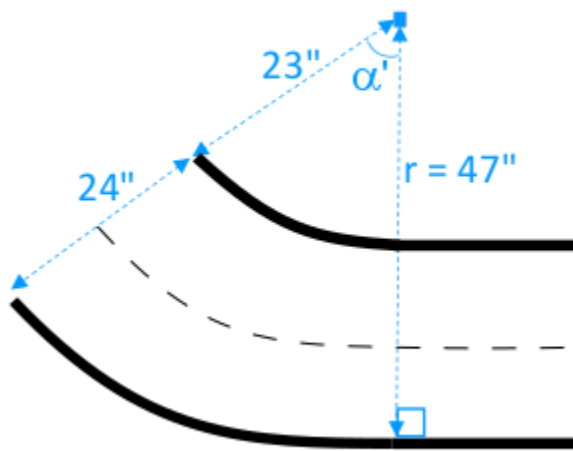
- [寸法の要件](#)
- [モデルのパフォーマンスに関する考慮事項](#)
- [トラックを構築するステップ](#)

寸法の要件

次の要件を満たしていれば、任意の形状のトラックを作成できます。

- 最小回転半径:

カーブのトラックでは、回転半径 (r) は、円の中心から外側の境界までを下图に示すように、測定します。



最小回転半径 (r_{\min}) はコーナーでのトラック回転角度 (α) によって異なり、次の制限に準拠する必要があります。

- トラックの回転角度が $\alpha \leq 90$ degrees の場合は、

$$r_{\min} \geq 25 \text{ inches}$$

30 インチをお勧めします。

- トラックの旋回角度が $\alpha > 90$ degrees の場合は、 α

$$r_{\min} \geq 30 \text{ inches.}$$

35 インチをお勧めします。

- トラックの幅、

トラックの幅 (w_{track}) には、次の制限に準拠する必要があります。

$W_{\text{track}} \geq 24 \pm 3 \text{ inches.}$

- **トラック表面:**

トラック表面は滑らかで均一な暗い色である必要があります。最小包含領域は、30 inches x 60 inches にする必要があります。

カーペット敷きや木製の床が適切です。[連動フォームまたはゴムパッド](#)は木材よりもシミュレーション環境としてマッチしますが、必須ではありません。コンクリートの床は、表面での光の反射が原因で問題が発生する可能性があります。

- **トラックの障壁**

必須ではありませんが、トラックは、高さ 2.5 フィート以上、トラックから 2 フィート以上離れていない、均一な色の障壁で囲むことをお勧めします。

モデルのパフォーマンスに関する考慮事項

トラックの構築方法は、トレーニングされたモデルの信頼性と性能に影響を与える可能性があります。以下は、独自のトラックを構築するときの考慮すべき要素です。

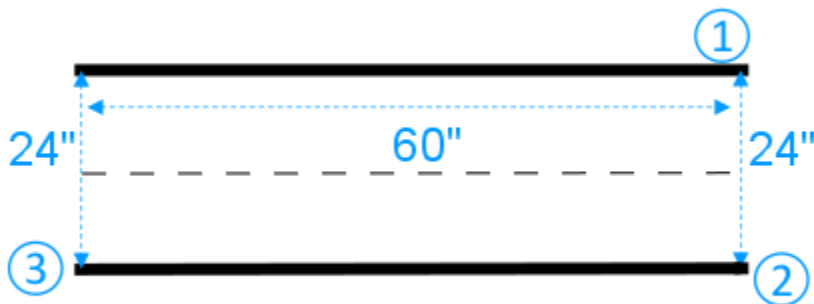
1. トラックの上や近くに白いオブジェクトを置かないでください。必要に応じて、トラックやその周辺から白いオブジェクトを取り除いてください。これは、シミュレーション環境でのトレーニングではトラック境界のみだけが白であると想定しているためです。
2. トラックの境界線には、清潔で連続したテープを使用してください。トラックの境界線が破損したり折れ曲がったりしていると、トレーニングを受けるモデルのパフォーマンスに影響を与える可能性があります。
3. トラックフロアとして反射面を使用しないでください。明るいライトからまぶしさを低減します。直線のエッジからのまぶしさは、オブジェクトまたは境界として誤って解釈される可能性があります。
4. トラックライン以外のラインマークの付いたトラックフロアを使用しないでください。このモデルは、トラックラインではないものをトラックの一部として解釈することがあります。
5. 背景オブジェクトから集中力をそぐものを減らすためにトラックの周りに障壁を置いてください。

トラックを構築するステップ

例として、最も基本的なシングルターントラックを使用します。Sカーブ、ループ、または AWS re:invent 2018 トラックなど、より複雑なトラックを作成するように指示を変更できます。

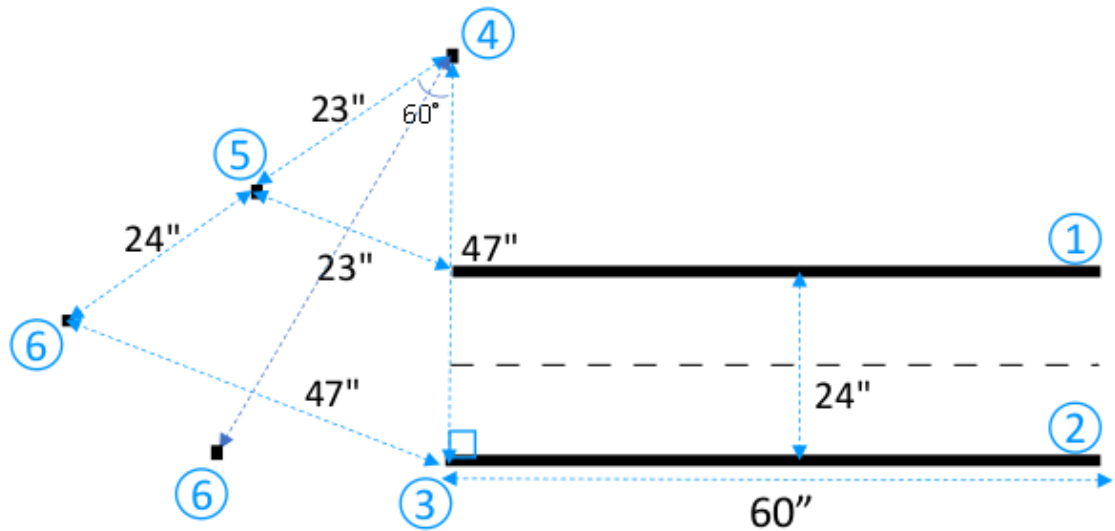
AWS DeepRacer のシングルターントラックの構築

1. トラックの直線部分を作成するには、以下の手順に従って図を参照してください。
 - a. 床の上に長さ 60 インチのテープを配置して、最初の境界を一直線にします (1)。
 - b. 巻尺を使用して、2 番目の境界線の 2 つの端点、(2) と (3) を見つけます。最初の境界の両端から 24 インチ離します。
 - c. 2 つの端点 (2) と (3) を接続するために、もう 1 の長さ 60 インチのテープを床に配置し、2 番目の境界を敷きます。



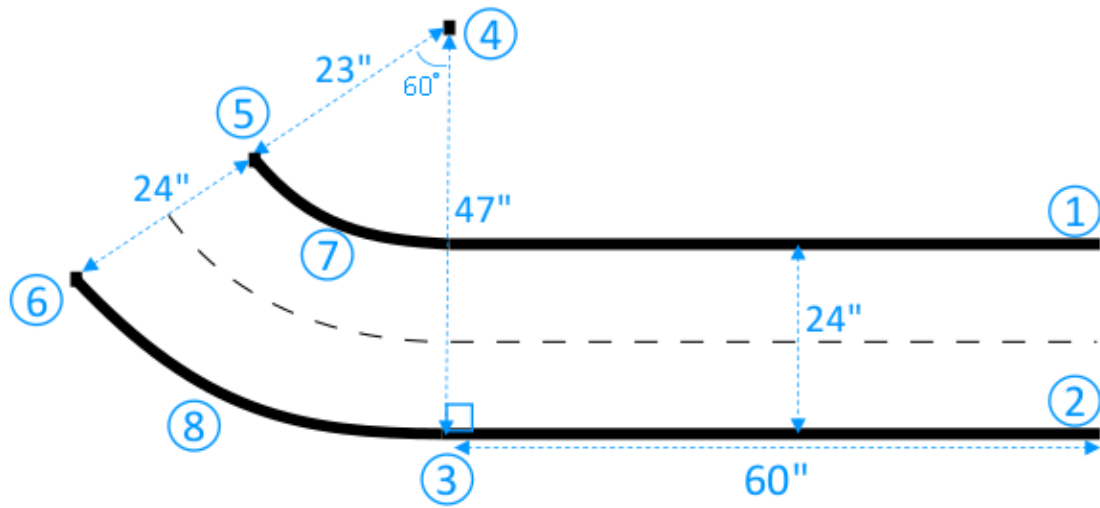
直線のトラックセグメントは、長さ 60 インチ、幅 24 インチです。寸法の要件を満たしている場合は、スペースに合わせて長さや幅を調整できます。

2. トラックを 60 度の角度で回転させるには、次の手順を実行して図を参照してください。
 - a. 巻き尺を使用して、回転半径 (4-3 または 4-6) の中心 (4) を見つけます。テープで中心に印を付けます。
 - b. 正三角形を描きます。3 辺が (3-4)、(4-6)、および (6-3)。

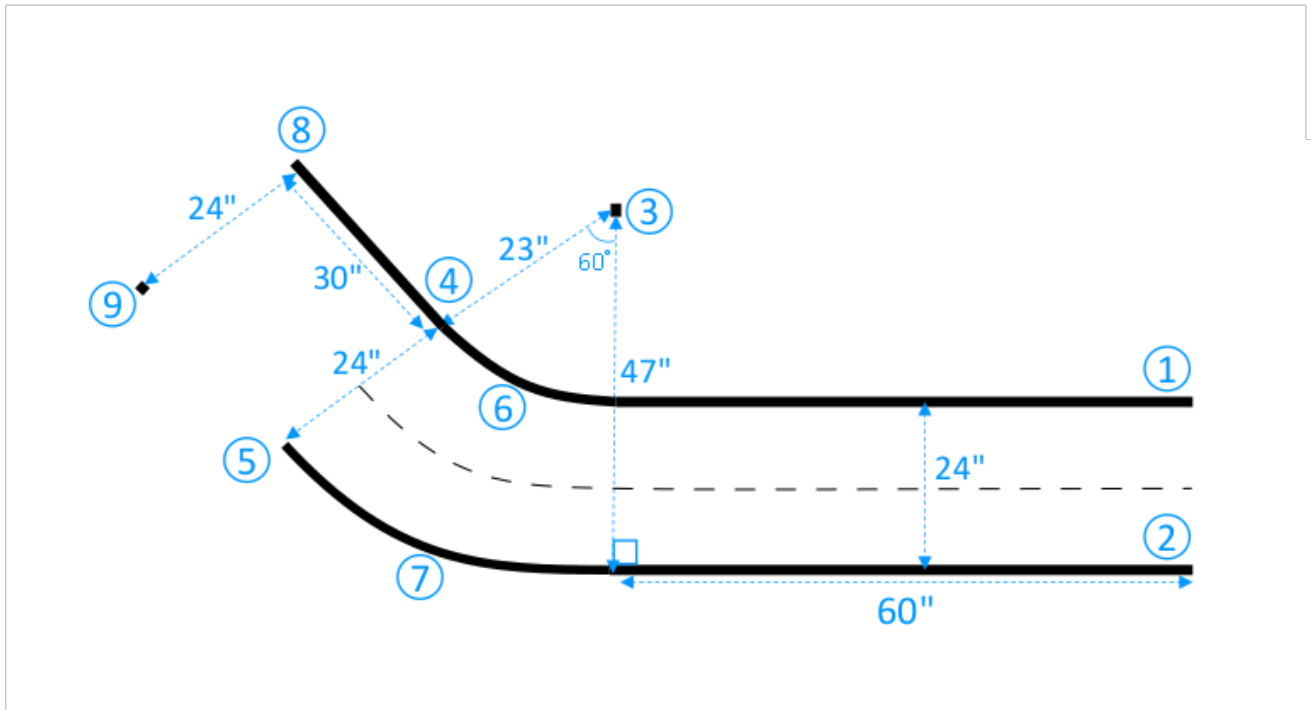


トラックに沿って60度回転させるには、正三角形(3-4-6)を使用して曲線トラックセグメントの2つの最終端点(5)と(6)の位置を決定します。別の角度のターンの場合は、分度器(または分度器アプリ)を使用して、曲線トラックセグメントの2つの最終端点(5)と(6)を見つけることができます。ステップ2の最小回転半径の要件が満たされている限り、回転半径の変動は許容されます。

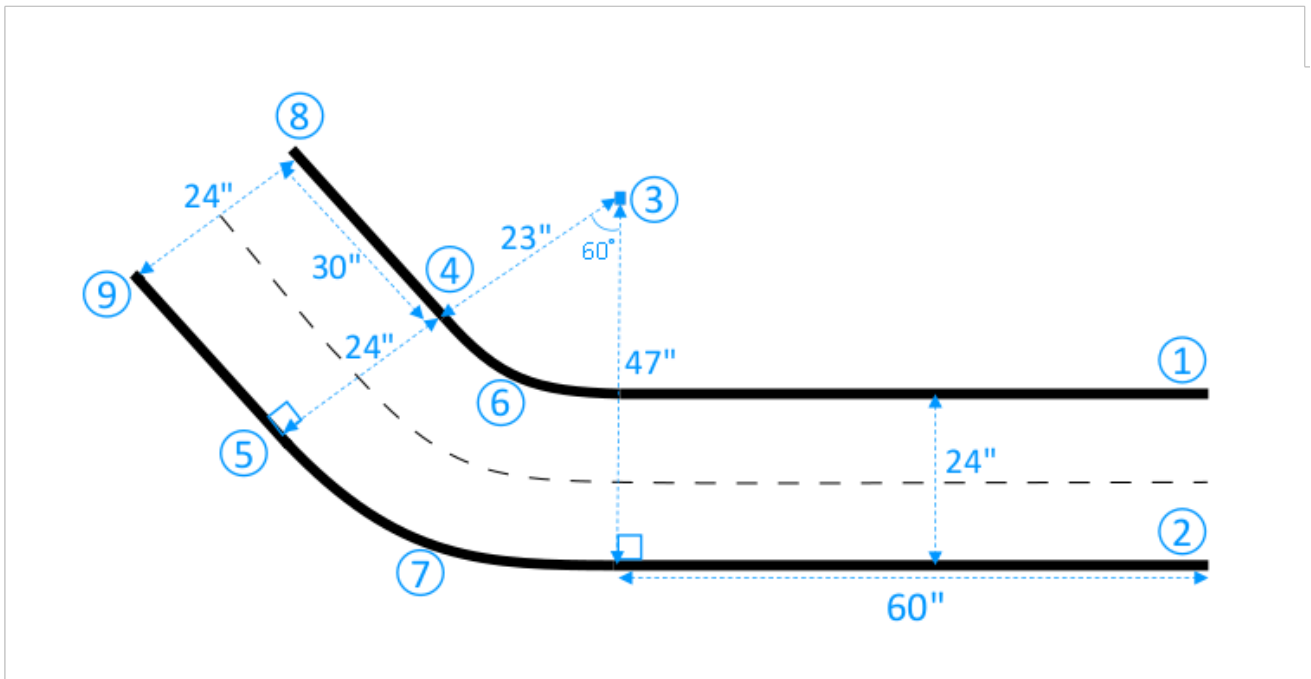
- c. たとえば、それぞれ4インチの小さなテープセグメントを床に配置し、曲線のボーダーセグメント(7)と(8)を敷き、それらを直線の境界で接続します。2つの曲線の境界線は平行である必要はありません。



3. 長さ 30 インチ、幅 24 インチの次の直線部分でトラックを拡張するには、次の手順に従います。
 - a. 長さ 30 インチのテープを床に置き、最初の境界線 (4-8) をエッジ (3-5) に対して垂直に配置します。



- b. 巻き尺を使用して、2 番目の枠線の終点 (9) を見つけます。お使いのスペースに合うように直線の長さをカスタマイズすることができます。
- c. もう一方の長さ 30 インチのテープを床に置き、2 番目の境界線 (5-9) をエッジ (3-5) に垂直に配置します。



2 番目の直線トラックセグメントは、長さ 30 インチ、幅 24 インチとします。寸法要件が満たされ、寸法が他のトラックセグメントと一致している場合は、スペースに合わせて長さと幅を調整できます。

4. 必要に応じて、長さ 4 インチのテープセグメントを切断してから、テープセグメントをトラックの中心に沿って 2 インチ離して配置し、破線を中心線を引きます。

これでシングルターントラックの構築は完了しました。走行可能なサーフェスと走行不可能なサーフェスをよりよく区別できるようにするには、オフトラックサーフェスに、オントラックサーフェスの色とは十分に対照的な色を塗る必要があります。安全性を確保するために、高さ 2.5 フィート以上、トラックから 2 フィート離れた均一な色の障壁でトラックを囲んでください。

トラックをより複雑な形状に拡張するための手順を適用できます。

AWS DeepRacer トラック設計テンプレート

次のトラックデザインテンプレートは、このセクションの [\[手順\]](#) に従って構築できる AWS DeepRacer トラックを示しています。

Note

このセクションには、事前にプリントされた利用可能なトラックのテンプレートで示されています。事前にプリントされたトラックを組み立てる方が、独自の素材でトラックを構築するよりも時間がかからず、プロセスも簡単です。事前にプリントされたトラックと障壁を使用することをお勧めします。事前にプリントされたトラックを購入するには、「[AWS DeepRacer ストアフロント](#)」を参照してください。

すべてのトラックについて、同じ色を再現するには、次の仕様に従ってください。

- 緑: PMS 3395C
- オレンジ: PMS 137
- 黒: PMS 432C
- 白: CMYK 0-0-2-9

これらのトラックは、表面を以下の材料でテストしました。

• ビニール

このトラックはまぶしさを低減するために、つや消し仕上げで 13 オンスのスクリームビニールにプリントされました。ビニールは一般的にカーペットよりも安く、良好なパフォーマンスを提供します。ビニールはカーペットほどの耐久性はありません。

• カーペット

このトラックはラテックスゴムの裏地付きの 8 オンスの昇華型で、表面をポリエステル加工したカーペットにプリントされました。カーペットは耐久性が高く、優れたパフォーマンスを提供しますが、コストが高くなります。

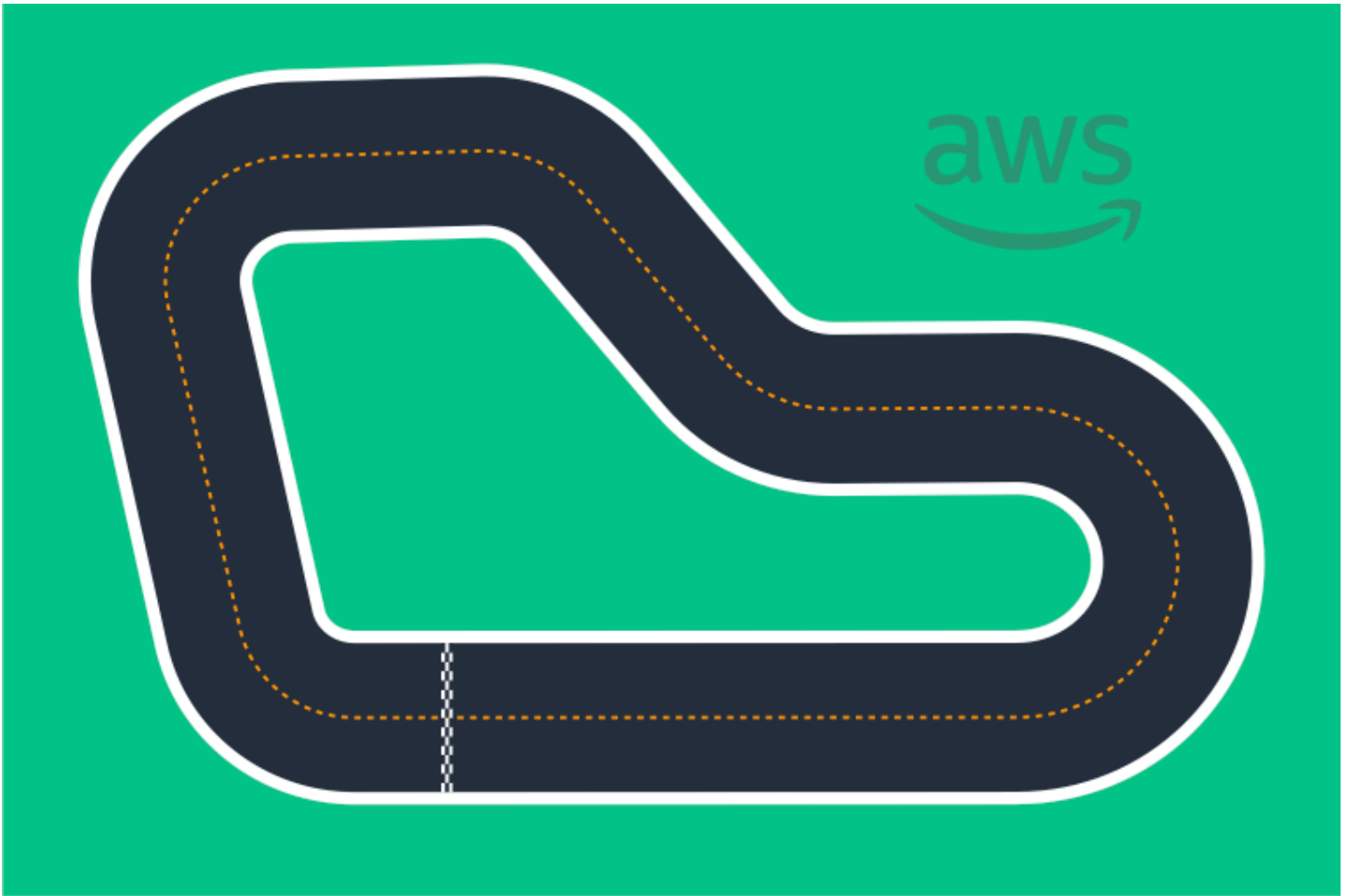
サイズが大きいため、トラックは単一の材料に簡単にはプリントできません。材料を接続する際は、トラックラインを揃えてください。

トピック

- [AWS DeepRacer A から Z までのスピードウェイ \(ベーシック\) トラックのテンプレート](#)
- [AWS DeepRacer Smile Speedway \(中級\) トラックのテンプレート](#)
- [AWS DeepRacer RL スピードウェイ \(上級\) トラックのテンプレート](#)
- [AWS DeepRacer シングルターントラックのテンプレート](#)
- [AWS DeepRacer S カーブトラックのテンプレート](#)
- [AWS DeepRacer ループトラックのテンプレート](#)

AWS DeepRacer A から Z までのスピードウェイ (ベーシック) トラックのテンプレート

AWS DeepRacer A から Z までのスピードウェイ (ベーシック) トラックは、AWS DeepRacer 史上最も人気のある物理的な競技用トラックです。もともと AWS re:invent 2018 でリリースされ、すべての AWS DeepRacer 物理コンペティショントラックのフットプリントが最も小さくなっています。事前にプリントされており、「[AWS DeepRacer ストアフロント](#)」で購入できます。

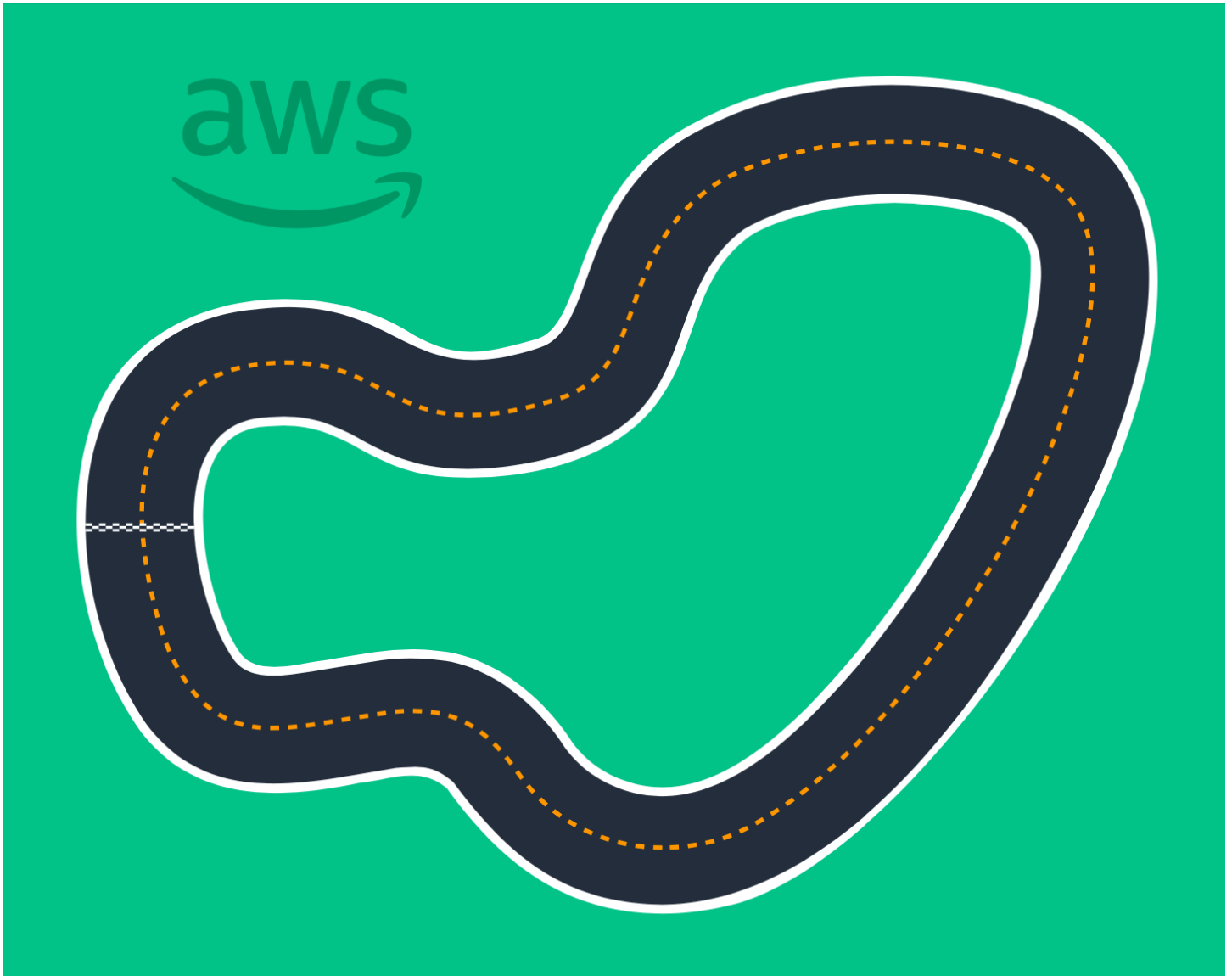


初心者イベントや初めてのレーサーには、このトラックをお勧めします。曲線や直線のさまざまなトラックがあり、初めてのレーサーや経験豊富なレーサーの両方にとって魅力的な挑戦となるでしょう。AWS DeepRacer A から Z までのスピードウェイ (ベーシック) トラックは、コンソールで利用できる仮想トラックを 1:1 で物理的に再現したものです。これにより、レーサーは仮想環境でモデルをトレーニングし、そのモデルを物理的な AWS DeepRacer デバイスにデプロイして、物理的なトラックでの自律レースを行うことができます。

独自の A から Z までのスピードウェイ (ベーシック) トラックをプリントまたは作成するには、この「[AWS DeepRacer A から Z までのスピードウェイ \(ベーシック\) ファイル](#)」をダウンロードしてください。

AWS DeepRacer Smile Speedway (中級) トラックのテンプレート

AWS DeepRacer Smile Speedway トラックは、もともと AWS DeepRacer チャンピオンシップ 2019 のトラックとしてリリースされました。事前にプリントされており、「[AWS DeepRacer ストアフロント](#)」で購入できます。



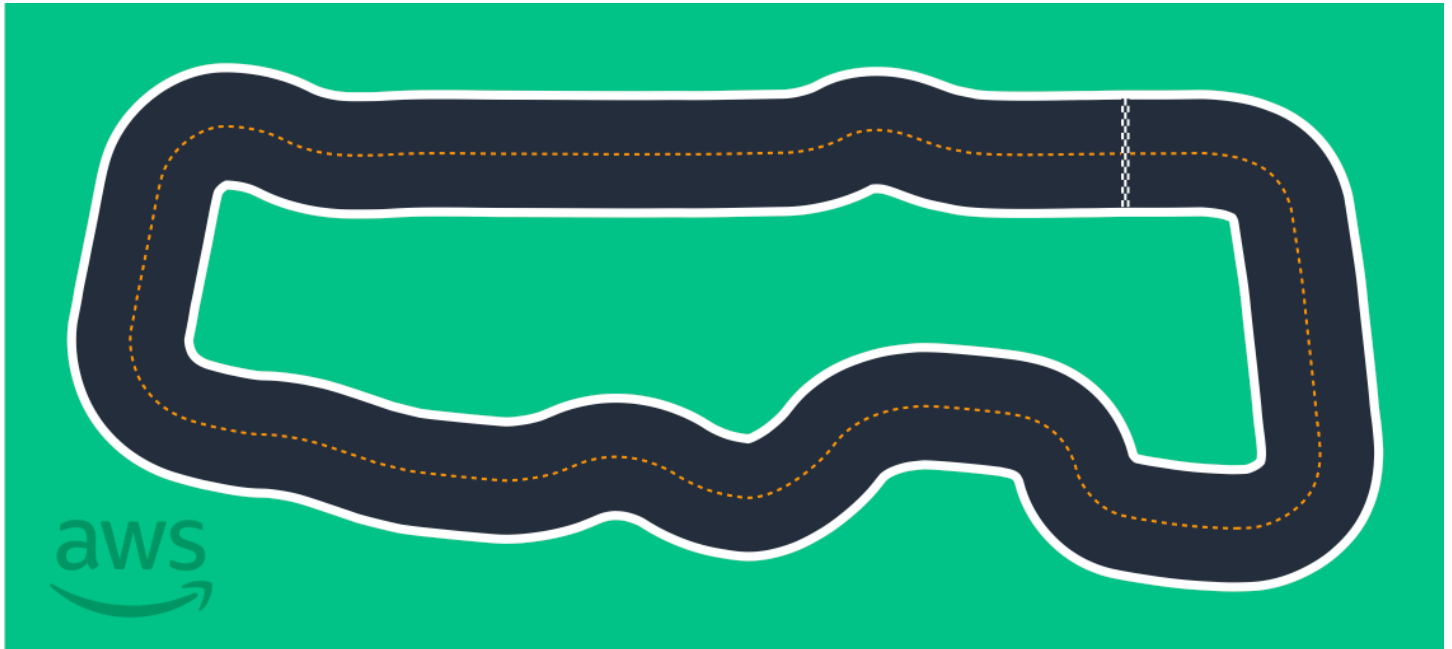
経験豊富なレーサーが参加するイベントや、物理的なスペースが広いイベントには、この中級トラックをお勧めします。コンソールで利用できる仮想トラックを 1:1 で物理的に再現したものです。これにより、レーサーは仮想環境でモデルをトレーニングし、そのモデルを物理的な AWS DeepRacer デバイスにデプロイして、物理的なトラックでの自律レースを行うことができます。

独自の AWS DeepRacer Smile Speedway (中級) トラックをプリントまたは作成するには、この [「AWS DeepRacer Smile Speedway \(中級\) トラックのファイル」](#) をダウンロードしてください。

AWS DeepRacer RL スピードウェイ (上級) トラックのテンプレート

AWS DeepRacer RL スピードウェイ (上級) トラック (別名 AWS DeepRacer サミットスピードウェイ) は、もともと 2022 年に AWS DeepRacer サミット向けにリリースされたもので、AWS

DeepRacer 史上最長の物理的なトラックです。事前にプリントされており、「[AWS DeepRacer スタートフロント](#)」で購入できます。

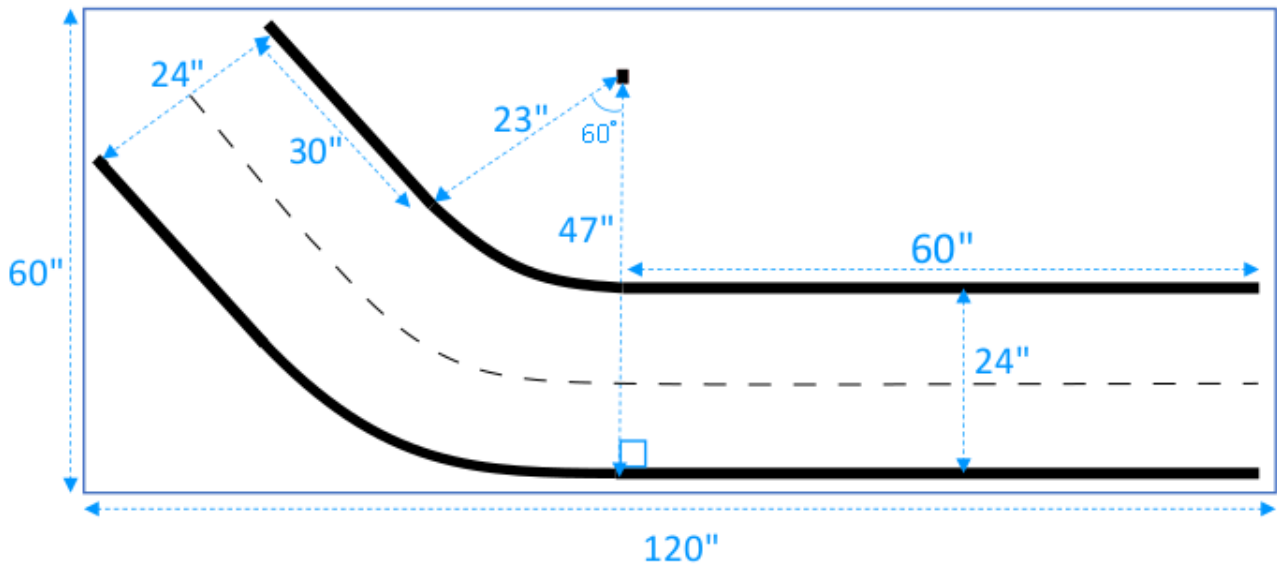


経験豊富なレーサーが集まるイベントには、AWS DeepRacer RL スピードウェイ (上級) トラックをお勧めします。直線で速く走ることを好むレーサーにとっては、挑戦しがいのあるコースです。AWS DeepRacer RL スピードウェイ (上級) トラックは、コンソールで利用できる仮想トラックを 1:1 で物理的に再現したものです。これにより、レーサーは仮想環境でモデルをトレーニングし、そのモデルを物理的な AWS DeepRacer デバイスにデプロイして、物理的なトラックでの自律レースを行うことができます。

独自の AWS RL Speedway (Advanced) トラックを印刷または作成するには、この [AWS DeepRacer RL Speedway \(Advanced\) トラックファイル](#) をダウンロードします。

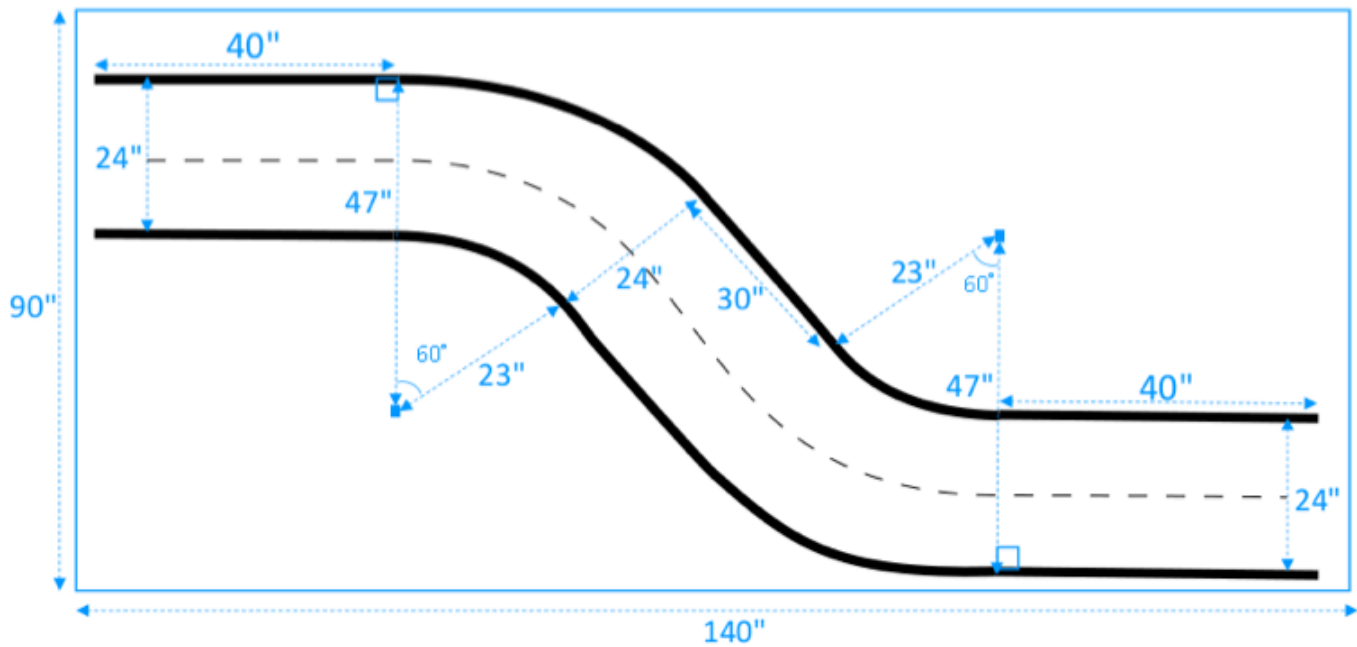
AWS DeepRacer シングルターントラックのテンプレート

この基本的なトラックのテンプレートは、曲線トラックセグメントで接続された 2 つの直線トラックセグメントで構成されています。このトラックでトレーニングされたモデルは、AWS DeepRacer 車両を直線で走行するか、または一方向に曲がるように走行します。



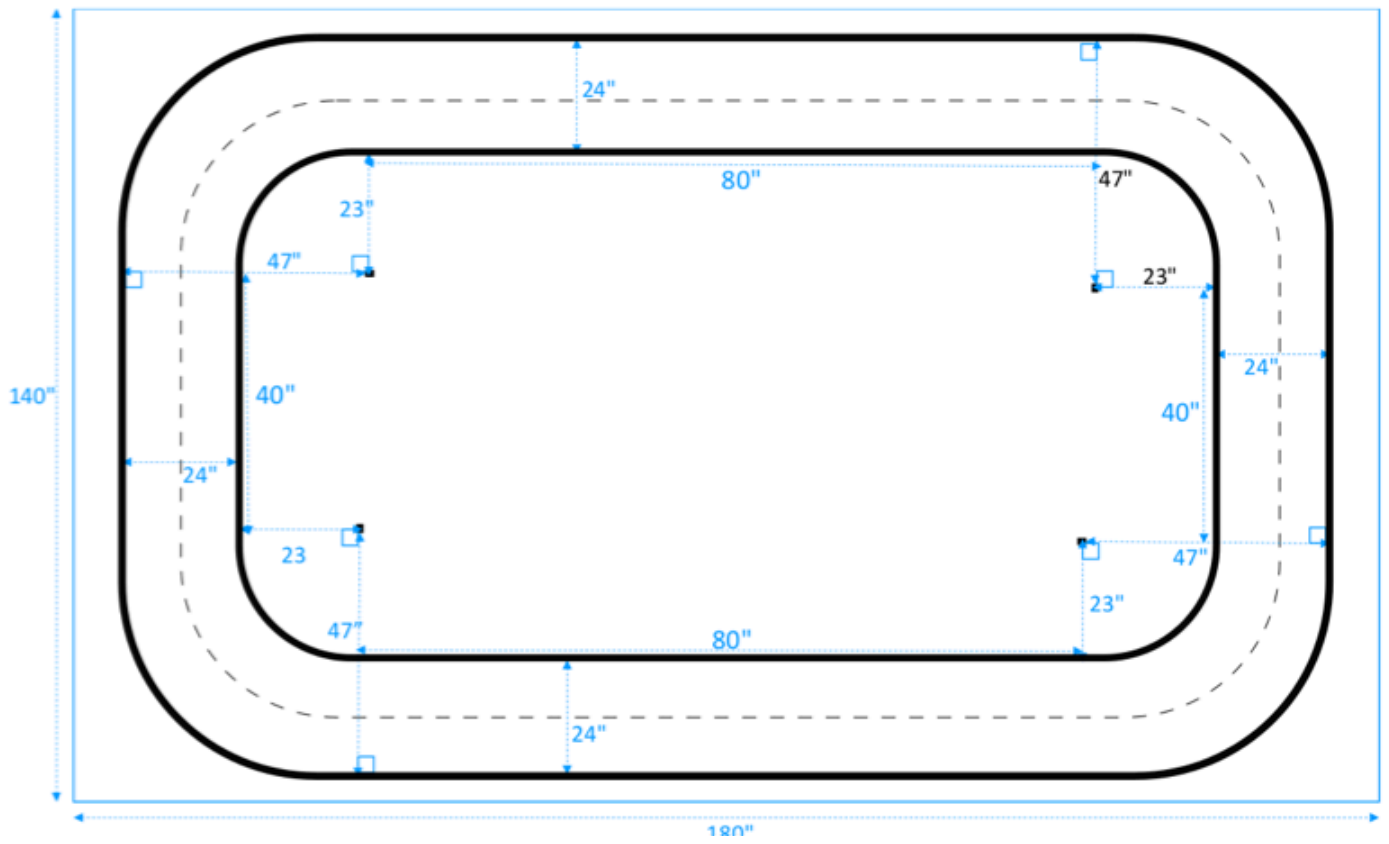
AWS DeepRacer S カーブトラックのテンプレート

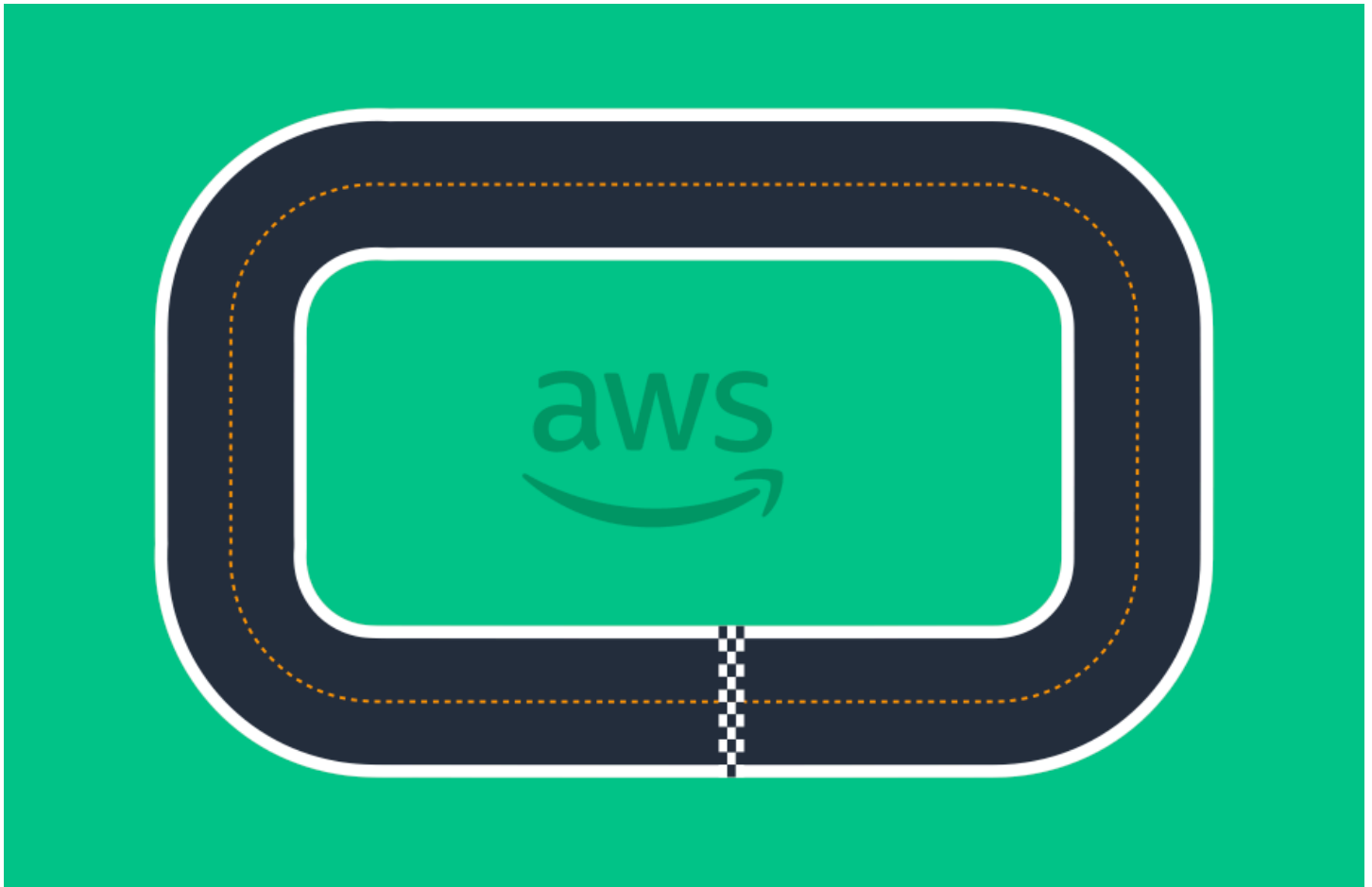
モデルは2方向にターンするように学習する必要があるため、このトラックはシングルターントラックよりも複雑です。簡単に最初のターンの後に反対方向に向きを変えることによってこのトラックにシングルターントラックの建設手順を簡単に拡張できます。



AWS DeepRacer ループトラックのテンプレート

この通常のループトラックは、90度のシングルターントラックを繰り返します。このトラック全体を敷設するためには、より広い包囲面積が必要になります。





AWS DeepRacer レースに参加する

シミュレーションで自分のモデルのトレーニングおよび評価ができれば、レースに参加してそのモデルのパフォーマンスを他のレーサーのものと比較することをお勧めします。レーシングは、モデルに関するフィードバックを受け取り、賞や賞品を獲得し、他の AWS DeepRacer のコミュニティの方々と出会い、スキルを習得して向上させることを学び、楽しい時間を過ごせる良い機会です。

レースは対面でもオンライン (バーチャル) でも可能で、バーチャルレースは同期的な LIVE レース、もしくは非同期的な [クラシック] にフォーマットできます。LIVE およびクラシックバーチャルレースは、個人的にあるいは公的に配信できます。

このセクションでは AWS DeepRacer League のバーチャルサーキット、もしくはコミュニティベースのバーチャルレースに参加する方法およびフォーマットに関する様々な選択肢について説明します。

AWS DeepRacer レースイベントの種類

イベントは、スポンサーまたは主催者によって分類できます。AWS DeepRacer League のレースイベントとコミュニティ型のレースイベントは両方とも、実際のトラック上で行われる対面でのレース、またはバーチャルのトラック上のオンラインのレースとして開催できます。

- AWS主催のレースイベント – 主催のレースイベント AWS は AWS DeepRacer League イベントと呼ばれ、すべての AWS DeepRacer ユーザーが利用できます。初めてのレーサーは、毎月開催される仮想レースに参加することでリーグとのジャーニーを始めることができます。レースにモデルを提出したレーサーはポイントを獲得し、国内およびリージョンでのシーズン順位を受け取ります。
- [コミュニティ主催のレースイベント] - AWS DeepRacer ユーザーが作成したレースイベントは、コミュニティレースイベントと呼ばれます。

オンライン AWS スポンサーまたはコミュニティスポンサーのレースへの参加

AWS DeepRacer コンソールを使用して、AWS DeepRacer League の仮想サーキットイベント、またはコミュニティ型のオンラインレースに参加できます。

- AWS DeepRacer ユーザーは、AWS DeepRacer League 仮想サーキットで受付中のすべてのオンラインレースに参加できます。
- コミュニティの仮想レースイベントには、招待されたユーザーだけがアクセスまたは参加できます。ユーザーは、レース主催者から送信された招待リンクを受信するか、または他のレース参加者によって転送された招待リンクを受信したときに招待されます。

トピック

- [the section called “ 仮想サーキットレースに参加する ”](#)
- [the section called “ コミュニティレースに参加する ”](#)
- [the section called “ LIVE レースに参加する ”](#)
- [the section called “ レースイベントの用語 ”](#)

AWS DeepRacer League の仮想サーキットレースに参加します。

このセクションでは、AWS DeepRacer コンソールを使用してトレーニング済みのモデルをレースに登録する方法について学びます。

AWS DeepRacer League 仮想サーキットに参加するには

1. [AWS DeepRacer コンソール](#)にサインインします。
2. メインのナビゲーションペインで [AWS AWS 仮想サーキット] を選択します。
3. [AWS 仮想サーキット] ページの [オープンレース] セクションで、[レースに参戦] を選択します。
4. AWS DeepRacer League のレースに初めて出場する場合は、[AWS DeepRacer League レーサー名] の中の [レーサー名] でエイリアスを登録してください。
5. [モデルを選択] の中の [モデル] リスト内から、使用したいモデルを選択します。自分のモデルがトラックの形状に対応できるようにトレーニングされていることを確認します。
6. AWS DeepRacer リーグイベントに初めて参加する場合、[リーグ要件] で [居住国] を選択します。居住国を選択し、最初のモデルを提出すると、レースシーズンの間、固定され、賞品が授与されるときに確認されます。次に、チェックボックスを選択して、利用規約に同意します。
7. [レースに参戦] を選択して提出を完了します。各レースの出場枠は 50 です。

モデルの提出後、AWS DeepRacer コンソールが評価を開始します。このプロセスには最大 10 分かかることがあります。

8. レースのページで、詳細を確認します。
9. レースのページで、自分のレーサー名の下に申請状況を確認します。
10. レースのページでリーダーボードのランキングを参照し、自分のモデルが他のモデルと比べてどの位置にいるかを確認します。

モデルが3回連続してトラックを完走しなかった場合、モデルはリーダーボードのランキングリストに表示されません。リーダーボードのランキングには、一番高い成績の提出が反映されます。また、自分の国やリージョンの他のレーサーの中で自分がどの順位にランクされているかを判断するため、国内およびリージョンのシーズン順位も表示されます。

モデルを送信したら、自分のモデルの報酬と反復処理の機能を改善してパフォーマンスを向上させてください。別のアルゴリズムまたはアクションスペースで新しいモデルをトレーニングすることもできます。学び、調整し、再度レースして、報酬を得るチャンスを増やしましょう。

AWS DeepRacer コミュニティレースに参加するには

Note

AWS DeepRacer コミュニティレースに参加するには、まず主催者からレースへのリンクを受け取る必要があります。

AWS DeepRacer レースに参加するための招待を受け取ったら、そのレースが LIVE またはクラシックタイプかを確認します。

クラシックレース

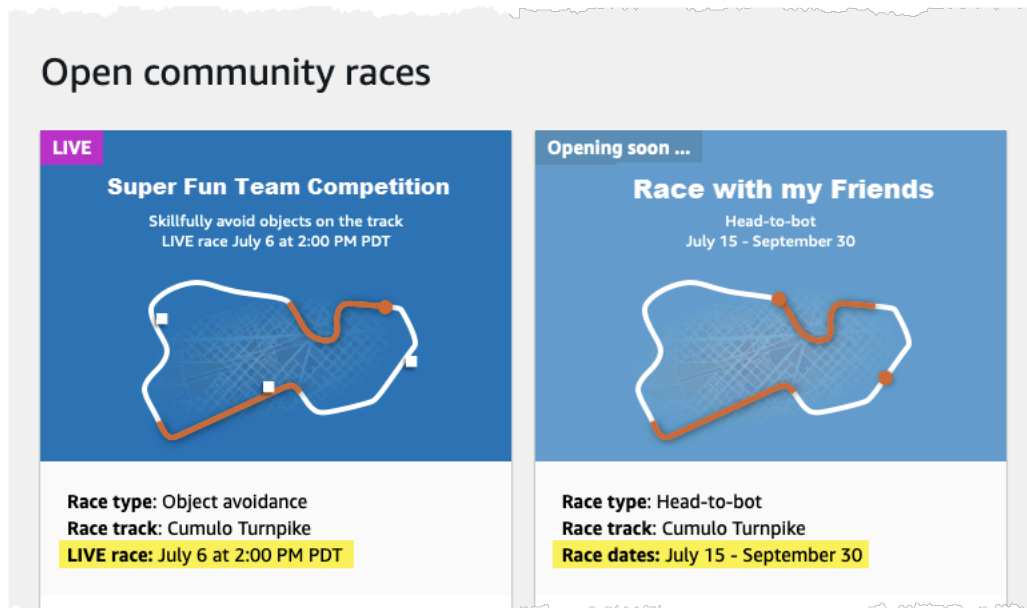
クラシックレースは、リアルタイムの対応を必要としない非同期型イベントです。招待リンクを使用すると、レースにモデルを提出してリーダーボードを閲覧することが可能になります。レースの開催期間内であればいつでも無制限にモデルを送信して、リーダーボードの首位獲得を目指することができます。送信されたモデルのクラシックレースの結果と映像は、リーダーボードのページでレース開始直後から確認できます。クラシックレースはすべてプライベートイベントです。

LIVE レース

LIVE レースは、他のレーサーとオンラインで集まり、リーダーボードで最速タイムを競うリアルタイムのレースイベントです。複数のモデルを送信することが可能ですが、提出期間が終わる前に最後に提出したモデルのみが使用されます。レース中は、インタラクティブなスピードコント

コースを試すことができます。これによりモデルの速度パラメータを一時的に上書きして、戦略的なリアルタイムの調整が可能となります。LIVE レースは、招待されたレーサーの間で個人的に放送することも、だれでも視聴できるように公開することもできます。

招待状に競技形式が指定されていない場合は、レースカードを確認してください。LIVE レースでは "LIVE" と表記され、同期しているイベントの日時が表示されます。クラシックレースでは、非同期競技の日程が表示されます。



レース参加者として AWS DeepRacer コミュニティレースに参加します。

を初めて AWS 利用し、AWS DeepRacer コミュニティレースへの参加の招待を受け取った場合は、「新しいユーザーとして参加するには」の手順に従います。活動中のコミュニティレースに招待され、過去に AWS DeepRacer レースに参加したことがある場合は、競技形式に合わせて [クラシックレースに参加するには] または [LIVE レースに参加するには] の手順に従います。

新しいユーザーとして参加するには

を初めて利用 AWS し、AWS DeepRacer コミュニティレースに参加するための招待リンクを受け取った場合は、リンクを選択して AWS DeepRacer コンソールに移動し、レースに参加する前に AWS アカウントにサインアップします。

AWS DeepRacer の新規ユーザー、または AWS DeepRacer レースへの初めての参加者は、下記のステップに従い AWS DeepRacer コンソール中のコミュニティレースに参加するステップに進みます。

新しいユーザーとして参加するには

1. [AWS DeepRacer コンソール](#)で AWS アカウントを作成します。
2. 登録とログインが完了後、レース主催者から届いているリンクを選択してレースのページを開きます。
3. [AWS DeepRacer レーサー名] の作成を求めるプロンプトが表示されたら、すべての AWS DeepRacer リーダーボードで、識別するために使用する名前を入力します。一度レーサー名を選択すると、変更することはできません。
4. レースの詳細ページで、[レースを開始する] を展開します。
5. 自動運転のために AWS DeepRacer モデルのトレーニングの簡単な紹介を受けるには、[RL の使用を開始する] を選択します。
6. AWS DeepRacer コンソール内のレース用に、モデルをトレーニングして評価します。

モデルのトレーニングの詳細については、「[最初の AWS DeepRacer モデルのトレーニング](#)」を参照してください。

7. [コミュニティレース] に移動します。
8. 招待されたレースを見つけます。レースカードで、[レースに参戦] を選択します。

AWS DeepRacer × New: DeepRacer LIVE enables in-console real-time virtual races. [Create your race now!](#)

AWS DeepRacer > Community races

Welcome to the 2021 AWS DeepRacer community races, racer1!

Create your own DeepRacer LIVE virtual race
Race on your own terms! Organize a private LIVE virtual event for your friends and peers.

Race for prizes and glory
Enter the DeepRacer League Virtual Circuit for a chance to win.

Get rolling with machine learning
Take this free 90 minute training and certification course to start your machine learning journey with DeepRacer.

Join an AWS DeepRacer community race
Learn more in the AWS DeepRacer Developer Guide.

Connect with the community
Ask questions, exchange tips, and share best practices with fellow racers.

Open community races [Manage races](#) [Create race](#)

6 hours to LIVE race

LIVE! LIVE! LIVE!

Skillfully avoid objects on the track
LIVE race July 8 at 12:00 AM PDT

Race type: Object avoidance
Race track: Cumulo Turnpike
LIVE race: July 8 at 12:00 AM PDT

Race entries open
racer1

Your rank --/-- Gap to fastest --

[Leaderboard](#) [Race again](#)

Model submitted: Tagris-terminator

1 day to LIVE race

College vs. Colleg...

Race against AWS bot cars
LIVE race July 9 at 12:00 AM PDT

Race type: Head-to-bot
Race track: Cumulo Turnpike
LIVE race: July 9 at 12:00 AM PDT

Race entries open
racer1

Your rank --/-- Gap to fastest --

[Leaderboard](#) [Enter race](#)

23 days remaining!

Super Team Time Fu...

Head-to-bot
July 7 - July 31

Race type: Head-to-bot
Race track: re:Invent 2018
Race dates: July 7 - July 31

Race entries open
racer1

Your rank 1/1 Gap to fastest +00:00.000

[Leaderboard](#) [Race again](#)

▶ **Completed races (18)**

- 自分のレースの競技形式に合わせて、[クラシックレースに参加する]または[LIVEレースに参加する]の手順に従います。

クラシックレースに参加する方法

- レース主催者から受け取ったリンクを選択します。[AWS DeepRacer コンソール](#)で自分のアカウントにまだサインインしていない場合は、サインインが要求されます。

2. AWS DeepRacer コンソールにサインインすると、リンクからレースページに移動します。レースページにはレースの詳細、リーダーボード、およびあなたのレーサーの情報が表示されます。[レースに参加] を選択します。

The screenshot shows the AWS DeepRacer console interface for a community race titled "Super Team Time Fun!". The page is divided into several sections:


- Left Sidebar:** Contains navigation links such as "Racing League", "Reinforcement learning", "Resources", and "Next challenge".
- Header:** Shows the breadcrumb "AWS DeepRacer > Community races > Super Team Time Fun!" and a red-circled "Enter race" button.
- Race details:**
 - Race hosting:** Classic race
 - Race type:** Head-to-bot
 - Race dates:** Start July 7, 2021 at 12:00 AM; End July 31, 2021 at 12:00 AM
 - Time zone:** UTC-0700 (Pacific Daylight Time) America/Los_Angeles
 - Competition track:** Inspired by Monza, re:Invent 2018 was the first Championship Cup track. This short, classic speedway remains a perennial rookie favorite. Length: 17.6 m (57.97') Width: 76 cm (30")
 - Rules:**
 - Ranking method: Total time
 - Style: Individual lap
 - Entry criteria: 3 consecutive laps
 - Resets: Unlimited resets
 - Off-track penalty: 3 seconds
 - Head-to-bot rules:**
 - Number of bot cars: 3 cars
 - Bot car speed: 0.75 m/s
 - Bot lane change: Disabled
 - Collision penalty: 3 seconds
- Leaderboard:** Titled "Super Team Time Fun! leaderboard", it shows a search bar and a table with columns for Rank, Racer, Time, Gap to 1st, Video, Off-track, and Collision. The current state is "No entries. Be the first to make it onto this leaderboard!".
- Start your engines:** A section titled "Train a model" with instructions: "To increase your chances of a good ranking, ensure you train a model type that matches the race type, and that your training setup (track and bots) mimics the race setup. Good luck in the race!". A "Train a model" button is present.

3. 「レースに参加する」ページ内で、[モデルの選択] からトレーニング済みモデルを選択し、[レースに参加] を選択します。

AWS DeepRacer > Community races > Super Team Time Fun! > Enter race

Enter race

Super Team Time Fun!

Race hosting Classic race	Competition track Inspired by Monza, re:Invent 2018 was the first Championship Cup track. This short, classic speedway remains a perennial rookie favorite. Length: 17.6 m (57.97') Width: 76 cm (30")	Rules Ranking method Style Entry criteria Resets Off-track penalty	Total time Individual lap 3 consecutive laps Unlimited resets 3 seconds
Race type Head-to-bot		Head-to-bot rules Number of bot cars Bot car speed Bot lane change Collision penalty	3 cars 0.75 m/s Disabled 3 seconds
Race dates Start July 7, 2021 at 12:00 AM End July 31, 2021 at 12:00 AM			
Time zone UTC-0700 (Pacific Daylight Time) America/Los_Angeles			

Choose model

Selection and submission
Submit your model to participate in the virtual race. Your time and rank will be displayed on the race leaderboard alongside other competitors.

Tagris-terminator	▲
asjdfhasdf	
dafdsfasdfasdf	
Tagris-terminator	
Fabulous-mud	
Action-Space-Activator	
Trin...	

Cancel Enter race

4. モデルがレース基準に照らして正常に評価された場合は、イベントのリーダーボードを見て、他の参加者に対して自分のモデルがどのようにランク付けされるかを確認してください。
5. 必要に応じて、[見る] を選択し、自分の車両のパフォーマンスのビデオを見るか、[評価ログをダウンロードする] をクリックして、生成された実績を詳しく確認します。


Submission successful! Watch your video. [View Video](#)

AWS DeepRacer > Community races > Super Team Time Fun!

Super Team Time Fun!

[Race again](#)

Race details

Race hosting Classic race	Competition track Inspired by Monza, re:Invent 2018 was the first Championship Cup track. This short, classic speedway remains a perennial rookie favorite. Length: 17.6 m (57.97') Width: 76 cm (30")	Rules Ranking method: Total time Style: Individual lap Entry criteria: 3 consecutive laps Resets: Unlimited resets Off-track penalty: 3 seconds
Race type Head-to-bot		Head-to-bot rules Number of bot cars: 3 cars Bot car speed: 0.75 m/s Bot lane change: Disabled Collision penalty: 3 seconds
Race dates Start July 7, 2021 at 12:00 AM End July 31, 2021 at 12:00 AM	Time zone UTC-0700 (Pacific Daylight Time) America/Los_Angeles	

Super Team Time Fun! leaderboard (1)

Search by racer alias

Rank	Racer	Time	Gap to 1st	Video	Off-track	Collision
1	racer1	01:47.821		Watch	12	2

Your best model
[Tagris-terminator](#)

Latest model submitted

Name
Tagris-terminator

Total lap time
01:47.821

Submission time
7/7/2021, 12:51:56 PM PDT

Status
✔ Completed 3 laps
[Watch video](#)

[Download evaluation logs](#)

- [再度レースする] を選択して、別のモデルを登録します。レースの開催期間内であればいつでも無制限にモデルを送信して、リーダーボードの首位獲得を目指すことができます。

LIVE レースに参加するには

- レース主催者から受け取ったリンクを選択します。[AWS DeepRacer コンソール](#)で自分のアカウントにまだサインインしていない場合は、サインインが要求されます。
- AWS DeepRacer コンソールにサインインすると、リンクからレースページに移動します。レースページにはレースの詳細とリーダーボードが表示されます。[レースに参加] を選択します。

AWS DeepRacer ×


- ▼ Racing League
 - AWS Virtual Circuit
 - Community races
 - Your racer profile
- ▼ Reinforcement learning
 - Get started
 - Your models
 - Your garage
- ▼ Resources
 - About the league [🔗](#)
 - Schedules & standings [🔗](#)
 - Rules & prizes [🔗](#)
 - Developer guide [🔗](#)
 - Tips & tricks [🔗](#)
 - Forum [🔗](#)
 - Community Slack channel [🔗](#)
 - Buy AWS DeepRacer [🔗](#)
- ▼ Next challenge
 - Try a robotics project New!
 - Try computer vision [🔗](#)
 - Try generative AI [🔗](#)

AWS DeepRacer > Community races > LIVE! LIVE! LIVE!


Enter race

LIVE! LIVE! LIVE!

Race details

<p>Race hosting LIVE race</p> <p>Race type Object avoidance</p> <p>LIVE race date Start on July 7, 2021 at 12:00 AM (PDT)</p>	<p>Competition track The Cumulo Tumpike shifts from high-speed straightaways to challenging corners. It requires a perfect storm of exceptional navigation skill and speed control. Length: 60 m (197') Width: 106 cm (42")</p> 	<p>Rules</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <tr> <td style="border-bottom: 1px dashed #ccc; padding-bottom: 2px;">Ranking method</td> <td>Best lap time</td> </tr> <tr> <td style="border-bottom: 1px dashed #ccc; padding-bottom: 2px;">Style</td> <td>Individual lap</td> </tr> <tr> <td style="border-bottom: 1px dashed #ccc; padding-bottom: 2px;">Entry criteria</td> <td>3 consecutive laps</td> </tr> <tr> <td style="border-bottom: 1px dashed #ccc; padding-bottom: 2px;">Resets</td> <td>Unlimited resets</td> </tr> <tr> <td style="border-bottom: 1px dashed #ccc; padding-bottom: 2px;">Off-track penalty</td> <td>3 seconds</td> </tr> <tr> <td style="border-bottom: 1px dashed #ccc; padding-bottom: 2px;">Time per racer</td> <td>3 minutes</td> </tr> </table> <p>Object avoidance rules</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <tr> <td style="border-bottom: 1px dashed #ccc; padding-bottom: 2px;">Number of obstacles</td> <td>4</td> </tr> <tr> <td style="border-bottom: 1px dashed #ccc; padding-bottom: 2px;">Collision penalty</td> <td>3 seconds</td> </tr> </table>	Ranking method	Best lap time	Style	Individual lap	Entry criteria	3 consecutive laps	Resets	Unlimited resets	Off-track penalty	3 seconds	Time per racer	3 minutes	Number of obstacles	4	Collision penalty	3 seconds
Ranking method	Best lap time																	
Style	Individual lap																	
Entry criteria	3 consecutive laps																	
Resets	Unlimited resets																	
Off-track penalty	3 seconds																	
Time per racer	3 minutes																	
Number of obstacles	4																	
Collision penalty	3 seconds																	


Racers (1)



racer1
Awaiting Submission


LIVE Race starts July 7 at 12:00 AM PDT

TUNE IN
Leaderboard results posted here as soon as the race starts



12:00 AM PDT

Calendar



heat-jr
Your rank
--/--

LIVE! LIVE! LIVE! leaderboard

Rank	Racer	Time	Gap to 1st	Video	Off-track	Collision
Live racing results in on 7/7/2021, 12:00:00 AM. Results from live racing will appear on leaderboard						

Start your engines

Train a model

To increase your chances of a good ranking, ensure you train a model type that matches the race type, and that your training setup (track and obstacles) mimics the race setup. Good luck in the race!

Train a model

- 「レースに参加する」ページ内で、[モデルの選択] からトレーニング済みモデルを選択し、[レースに参加] を選択します。



AWS DeepRacer > Community races > LIVE! LIVE! LIVE! > Enter race

Enter race

LIVE! LIVE! LIVE!

Race hosting

LIVE race

Race type

Object avoidance

LIVE race date

Start on July 7, 2021 at 12:00 AM (PDT)

Competition track

The Cumulo Turnpike shifts from high-speed straightaways to challenging corners. It requires a perfect storm of exceptional navigation skill and speed control. Length: 60 m (197') Width: 106 cm (42")

**Rules**Ranking method

Style

Entry criteria

Resets

Off-track penalty

Time per racer

Best lap time

Individual lap

3 consecutive laps

Unlimited resets

3 seconds

3 minutes

Object avoidance rules

Number of obstacles

Collision penalty

4

3 seconds

Choose model

Selection and submission

Submit your model to participate in the virtual race. Your time and rank will be displayed on the race leaderboard alongside other competitors.

Fabulous-mud ▲

asjdfhasdf

dafdsfasdfasdf

Tagris-terminator

Fabulous-mud

Action-Space-Activator

Cancel

Enter race

4. モデルがレース基準に照らして正常に評価された場合は、イベントのリーダーボードを見て、他の参加者に対して自分のモデルがどのようにランク付けされるかを確認してください。
5. オプションとして、LIVE レースの場合は [カレンダー] をクリックして、LIVE レースイベントをカレンダーに追加します。
6. [再度レースする] を選択して、別のモデルを入力します。複数のモデルを送信することが可能ですが、提出期間が終わる前に最後に提出したモデルのみが使用されます。

AWS DeepRacer LIVE レースに参加する

Note

LIVE レースのスタート時間の少なくとも1時間前にモデルを送信してください。複数のモデルを登録することが可能ですが、提出期間が終わる前に最後に送信したモデルのみが使用されます。

開始する前に

- Chrome または Firefox のブラウザを使用してください (ブラウザが最新であることを確認してください)
- 仮想プライベートネットワーク (VPN) を使用している場合、VPN を切断してください
- 追加のタブをすべて閉じます

LIVE レースに参加するには

1. [AWS DeepRacer コンソール](#)にサインインします。
2. モデルをまだ送信していない場合は、参加したいレースのレースカードを見つけて、[LIVE レースに進む] を選択します。

AWS DeepRacer × **New: DeepRacer LIVE enables in-console real-time virtual races. [Create your race now!](#)**

AWS DeepRacer > Community races

Welcome to the 2021 AWS DeepRacer community races, racer1!

Official-DBS-DeepRacer-League

Race details
Time trial
 This AWS DeepRacer League is open to anyone in the bank! Checkout what's happening, how to get points and rewards, training calendar and etc at: <https://go.db.com/deepracer>
 Time remaining: **64 days left to race**

Release 2018
 Length | 17.6 mi (283)
 Road width | 76 cm (30")

Rules
 Handling method | Single lap time
 Style | Individual lap
 Entry criteria | 1 lap
 Resets | No resets

Racer alias
 davisia

Your rank: **--/1103**

Official-DBS-DeepRacer-League

Rank	Racer	Time	Gap to 1st	Video
1	RayG	00:07.635		Watch
2	Klemmizian	00:07.866	+00:00.231	Watch

Start your engines

Feedback | English (US)

Open community races Manage races Create race

LIVE

Race with Friends

Skillfully avoid objects on the track
LIVE race July 7 at 7:10 PM PDT

Race type: Object avoidance
Race track: Cumulo Turnpike
LIVE race: July 7 at 7:10 PM PDT

Good luck today!
racer1

Your rank: --/-- Gap to fastest: --

Leaderboard **Go to LIVE race**

5 hours to LIVE race

LIVE! LIVE! LIVE!

Skillfully avoid objects on the track
LIVE race July 8 at 12:00 AM PDT

Race type: Object avoidance
Race track: Cumulo Turnpike
LIVE race: July 8 at 12:00 AM PDT

Race entries open
racer1

Your rank: --/-- Gap to fastest: --

Leaderboard Race again

Model submitted: Tagris-terminator

1 day to LIVE race

College vs. Colleg...

Race against AWS bot cars
LIVE race July 9 at 12:00 AM PDT

Race type: Head-to-bot
Race track: Cumulo Turnpike
LIVE race: July 9 at 12:00 AM PDT

Race entries open
racer1

Your rank: --/-- Gap to fastest: --


Leaderboard Enter race

- 「レース」ページで [レースに参加する] を選択します。
- 「レースに参加する」ページの [モデルの選択] で、ドロップダウンメニューから送信するモデルを選択し、[レースに参加] を選択します。

AWS DeepRacer > Community races > Race with Friends > Enter race

Enter race

Race with Friends

Race hosting LIVE race	Competition track The Cumulo Tumpike shifts from high-speed straightaways to challenging corners. It requires a perfect storm of exceptional navigation skill and speed control. Length: 60 m (197') Width: 106 cm (42")	Rules <table border="1"> <tr> <td>Ranking method</td> <td>Best lap time</td> </tr> <tr> <td>Style</td> <td>Individual lap</td> </tr> <tr> <td>Entry criteria</td> <td>1 consecutive lap</td> </tr> <tr> <td>Resets</td> <td>Unlimited resets</td> </tr> <tr> <td>Off-track penalty</td> <td>3 seconds</td> </tr> <tr> <td>Time per racer</td> <td>3 minutes</td> </tr> </table>	Ranking method	Best lap time	Style	Individual lap	Entry criteria	1 consecutive lap	Resets	Unlimited resets	Off-track penalty	3 seconds	Time per racer	3 minutes
Ranking method	Best lap time													
Style	Individual lap													
Entry criteria	1 consecutive lap													
Resets	Unlimited resets													
Off-track penalty	3 seconds													
Time per racer	3 minutes													
Race type Object avoidance		Object avoidance rules <table border="1"> <tr> <td>Number of obstacles</td> <td>4</td> </tr> <tr> <td>Collision penalty</td> <td>3 seconds</td> </tr> </table>	Number of obstacles	4	Collision penalty	3 seconds								
Number of obstacles	4													
Collision penalty	3 seconds													
LIVE race date Start on July 7, 2021 at 7:10 PM (PDT)														

Choose model

Selection and submission
Submit your model to participate in the virtual race. Your time and rank will be displayed on the race leaderboard alongside other competitors.

- Tagris-terminator
- asjdfhasdf
- dafdsfasdfasdf
- Tagris-terminator**
- Fabulous-mud
- Action-Space-Activator

Cancel **Enter race**

- 「レース」ページで [LIVE レースに進む] を選択します。
- 「ライブレース」のページで、待機メッセージが表示されます。レース主催者から提供されたカンファレンスブリッジに移動します。

Welcome to Race with Friends LIVE!

Your race organizer is prepping the race. When it starts, look for your racer alias in the COMING UP section under the LEADERBOARD to find your live race time. If you need assistance, contact your race organizer.

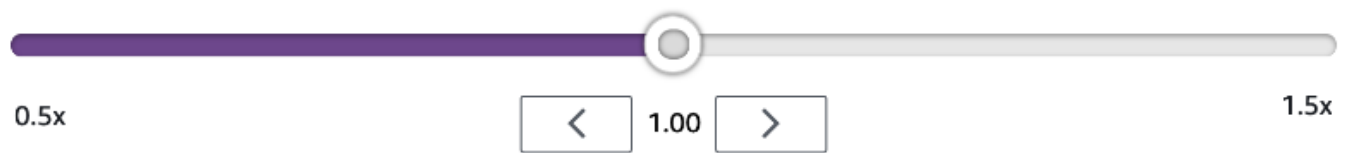
[Back to leaderboard details](#)

- レース主催者に話しかけると、レースのルールを確認し、質問に答えてくれます。
- リーダーボードの下の [カミングアップ] セクションでライブレースの開始時間をチェックし、レース主催者が自分の出番を伝えるまで準備を整えてください。
- 自分の番になると、レース主催者がレースをスタートさせるとき、コンソールに「10、9、8、7、6...」のカウントダウンアニメーションが表示されます。スタートを切った後、オプションのスピードコントロールにアクセスできます。モデルのスピードを調節する重

要な瞬間を選択するには、スピードコントロール機能进行操作するには、次の3つの方法があります。

- コンピュータのマウスでスライダーをドラッグします。
- またはコンソールで </> 矢印ボタンを選択します。
- スライダーノブを選択してスライダーを有効にし、# と # のキーボードの矢印キーを使用します。

Speed control



- モデルの速度パラメータの使用に戻るには、レーサーは乗数を1にリセットします。
- レース中に、LIVE レースのビデオオーバーレイをチェックして、パフォーマンスを最適化してください。トラックマップオーバーレイは、あなたのペースに応じて色が変化する3つのセクターに分かれています。レーサーが個人ベストを記録したセクションは緑色で示され、最も遅いセクターは黄色で示され、セッションベストの記録は紫色で示されます。レーサーは、ベストラップタイム、残り時間 (m/s)、リセット、現在のラップタイムを詳述した統計情報も検索します。



Track map overlay key:

- Green - Personal best
- Yellow - Slowest sector
- Purple - Session best

- コンソールにチェッカーフラッグのアイコンが表示されたら、レースが終了します。スピードコントロールが無効になり、ビデオ画面でレースのリプレイが開始されます。単独のベストラップタイムがリーダーボードにランク付けされます。

AWS DeepRacer コミュニティレースに参加するには

コミュニティレースとは、AWSから公式に資金提供を受けていない AWS DeepRacer ユーザーが主催するレースのことです。

レース招待リンクを共有して同僚やクラスメート、友人を招待することができます。

学生向けのレースを開催する場合は、「[AWS DeepRacer Student の教育者向けツール](#)」を参照してください。

トピック

- [the section called “レースを作成する \(クイックスタート\)”](#)
- [the section called “レースをカスタマイズする”](#)
- [the section called “LIVE レースで走行する”](#)
- [the section called “レースの管理”](#)
- [the section called “レースイベントの用語”](#)

仮想コミュニティレース: クイックスタートガイド

デフォルトのコミュニティレースの設定を使用して、仮想レースをすばやくセットアップできます。すべてのオプションについて学ぶ準備ができたなら、「[the section called “レースをカスタマイズする”](#)」を参照してください。

仮想レースを作成する前に、クラシックまたは LIVE レースはグループに最適であるかどうかを考え、ライブレースの場合はプライベートまたはパブリックにシェアするかどうかを選択します。

クラシックレース

クラシックレースは、リアルタイムの対応を必要としない非同期型イベントです。レースにモデルを提出し、リーダーボードを表示するには、参加者は招待リンクを受け取る必要があります。レーサーは、日付範囲内でも無制限のモデルを提出して、リーダーボードに登ることができます。速度コントロールは使用できません。提出されたモデルのクラシックレースの結果と映像は、リーダーボードのページでレース開始直後から確認できます。クラシックレースはすべてプライベートイベントです。

LIVE レース

LIVE レースは、1つのプライベートビデオ会議を容易にする1人のレース主催者による小規模なイベントから、主催者、コメンテーター、放送局の小規模なチームが公的に放送する大規模なイベントまでの範囲で行われる、設定された時間に行われるシンクロナイイベントです。モデル提出のドアはいつでも開閉できるので、レーサーに期限を知らせてください。参加者は複数のモデルを申告できますが、ドアを閉める前に最後に申告したモデルのみレースに参加できます。LIVE レースの実況中、キューに入っている参加者は、インタラクティブなスピードコントロールを使用して、自分の順番のときにモデルの競争力を高めることができます。LIVE レースの参加者は、レースにモデルを提出するための招待リンクも受け取らなければなりません。イベントを招待された参加者のみに非公開に放送するか、TwitchなどのLIVE ストリーミングサービスを使って公開するかを選択できます。詳細については、「[the section called “LIVE レースを放送する”](#)」を参照してください。

コミュニティレースの作成を開始するには

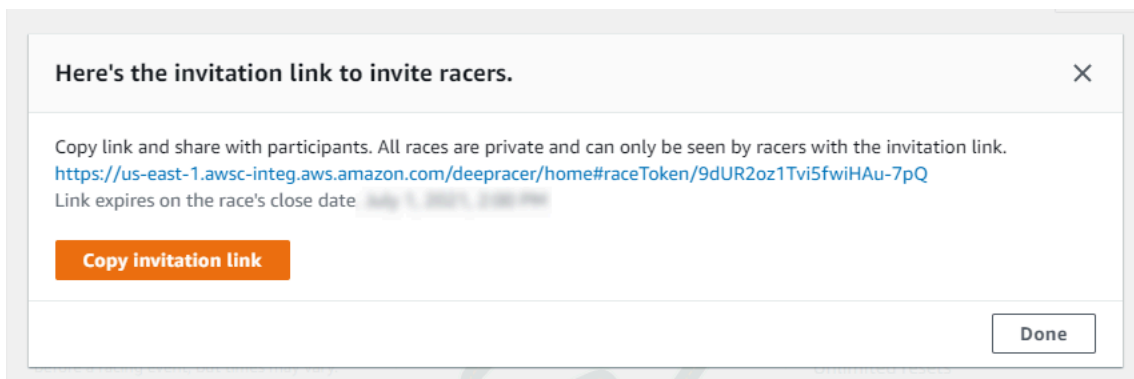
1. [AWS DeepRacer コンソール](#)を開きます。
2. [コミュニティレース] を選択します。
3. コミュニティレース で、[レースを作成] を選択します。

The screenshot shows the AWS DeepRacer console interface. On the left is a navigation sidebar with categories like 'Racing League', 'Reinforcement learning', 'Resources', and 'Next challenge'. The main content area is titled 'Community races' and includes a welcome message, a large image of a DeepRacer robot, and several links for creating and joining races. At the bottom, there is a section for 'Open community races' with three race cards. A red circle highlights the 'Create race' button in the top right corner of the 'Open community races' section.

- 「レースの詳細」ページで、[クラシックレース] 競技形式を選択します。これにより、設定した時間内にゲストが独自のスケジュールで参加できるようになります。あるいは LIVE レースで、リアルタイムイベントとして非公開または公開で放送できます。

クラシックレースの作成を続ける方法

- レースタイプを選択します。タイムトライアルからオブジェクトの回避へ、そして更に Head-to-bot へと、レースの種類は複雑さを増します。初めてのレーサーには、タイムトライアルをお勧めします。タイムトライアルレースでは、カメラが 1 台しか必要ないためセンサー構成が簡単になり、このタイプのレース用にトレーニングされた強化学習 (RL) モデルはより迅速に行われます。レースの詳細については、「[タイムトライアル、オブジェクト回避、およびヘッドツーボットレースのための AWS DeepRacer トレーニングを調整する](#)」を参照してください。
- レースには、オリジナルでわかりやすい名前を入力します。
- イベントの開始日時を 24 時間形式で指定します。AWS DeepRacer コンソールは、タイムゾーンを自動的に認識します。クラシックレースの場合は、終了日時も入力します。LIVE レースのデフォルト期間は 4 時間です。より長いレースをスケジュールするには、カスタマーサポートにお問い合わせください。LIVE レースが早期に終了した場合、実行するアクションはありません。
- デフォルトのレース設定を使用するには、[次へ] を選択してください。
- レースの詳細を確認するには、「レースの詳細」ページで、レースの仕様を確認してください。設定を変更するには、「レースの詳細」ページで、[編集] または [戻る] に戻ります。招待リンクを取得する準備ができたなら、[送信] を選びます。
- レースを共有するには、ポップアップから [招待リンクのコピー] をコピーし、メール、テキストメッセージ、お気に入りのソーシャルメディア アプリケーションにペーストします。クラシックレースは全て非公開であり、招待リンクのあるレーサーのみ見ることができます。リンクはレースの終了日に期限切れになります。



- [完了] を選択します。「レースを管理する」ページが表示されます。

- クラシックレースのタイムフレームが近づいたら、「リーダーボードの詳細」ページの [レーサー] で、誰がモデルに参入したのか、誰がまだそうする必要のあるのかをメモしておきましょう。

LIVE レースの作成を続ける方法

- レースタイプを選択します。タイムトライアルからオブジェクトの回避へ、そして更に Head-to-bot へと、レースの種類は複雑さを増します。初めてのレーサーには、タイムトライアルをお勧めします。タイムトライアルレースでは、カメラが 1 台しか必要ないためセンサー構成が簡単になり、このタイプのレース用にトレーニングされた強化学習 (RL) モデルはより迅速に行われます。レースの詳細については、「[タイムトライアル、オブジェクト回避、およびヘッドツーボットレースのための AWS DeepRacer トレーニングを調整する](#)」を参照してください。
- レースには、オリジナルでわかりやすい名前を入力します。
- イベントの開始日時を 24 時間形式で指定します。AWS DeepRacer コンソールは、タイムゾーンを自動的に認識します。クラシックレースの場合は、終了日時も入力します。LIVE レースのデフォルト期間は 4 時間です。より長いレースをスケジュールするには、カスタマーサポートにお問い合わせください。LIVE レースが早期に終了した場合、実行するアクションはありません。
- デフォルトのレース設定を使用するには、[次へ] を選択してください。
- レースの詳細を確認するには、「レースの詳細」ページで、レースの仕様を確認してください。設定を変更するには、「レースの詳細」ページで、[編集] または [戻る] に戻ります。招待リンクを取得する準備ができたなら、[送信] を選択します。
- 「<Your Race Name>」のページで、[招待] を選択してレースをシェアしてください。

The screenshot shows the AWS DeepRacer interface for managing a race. The left sidebar contains navigation options like 'Racing League', 'Reinforcement learning', and 'Resources'. The main content area is titled 'TestLiveRace' and has tabs for 'Race details', 'Invitation', and 'Racers'. The 'Invitation' tab is active, showing an invitation link, a 'Reset invitation link' button, and a suggested email template with a 'Copy' button.

- [招待の詳細] で招待リンクの [コピー] を選択し、メール、テキストメッセージ、お気に入りのソーシャルメディアアプリケーションにペーストします。
- 必要に応じて、推奨メールテンプレートの横にある [コピー] を選択します。賞品、モデル提出時間枠、そしてレーサーが出会ってレースの準備をする会議ブリッジのリンクを記入してください。

LIVE レースは非公開であり、公開放送を選択しない限り、招待リンクのあるレーサーのみ見ることができます。詳細については、「[the section called “LIVE レースを放送する”](#)」を参照してください。リンクは、レース終了日の午前 12:00 に期限切れになります。

- [詳細] タブを選択します。
- 「レースの詳細」で、LIVE レースを放送するためのオプションに注意してください。レースを公開するか非公開かを決定したら、AWS DeepRacer League チームが作成したプレイブックを使用して開始してください。[ブロードキャストの表示] ボタンを使用すると、LIVE レースイベントのページが表示され、コメンテーターストリームのカットアウトを含むブランドグラフィックオーバーレイで使用できます。
- LIVE レースの日程が近づいたら、「<Your Race Name>」ページの [招待] タブで、誰がモデルに参入したのか、誰がまだそうする必要があるので書き留めておいてください。

選択したレーストラックを変更するには、[\[コミュニティレースの管理\]](#) の [レースの詳細の編集] で、レースの説明を追加し、ランキング方法を選択し、レーサーが許可されるリセットの数を決定

し、RLモデルが自分のレースの資格を得るために完了しなければならない最小ラップ数を決定し、トラック外ペナルティを設定し、その他のレースの詳細をカスタマイズしてください。

レースをカスタマイズする

グループに合わせたレースを作成するには、「レースの詳細」ページの [レースのカスタマイズ] で作成します。タイムトライアルレースの設定は、オブジェクト回避と head-to-bot レースにも適用されますが、オブジェクト回避と head-to-bot のレースタイプには、イベントゴールに合わせて特別に調整されたレース環境を作成するためのコントロールを提供する追加設定があります。

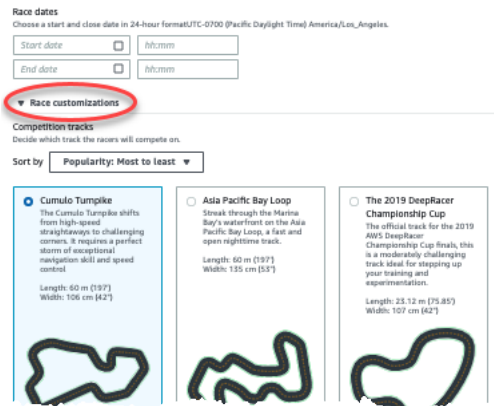
レースをカスタマイズするには

1. [\[AWS DeepRacer コンソール\]](#) を開きます。
2. [\[コミュニティレース\]](#) を選択します。
3. コミュニティレースで、[\[レースを作成\]](#) を選択します。

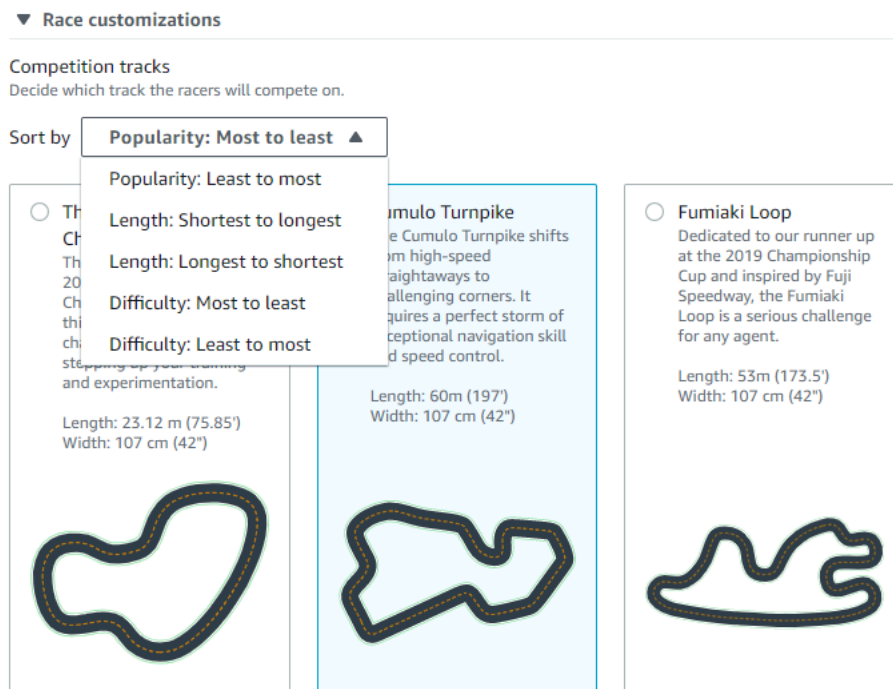
The screenshot shows the AWS DeepRacer console interface. On the left is a navigation sidebar with categories: Racing League, Reinforcement learning, and Resources. The main content area is titled 'Welcome to the 2021 AWS DeepRacer community races'. It includes a large image of a DeepRacer robot. Below the image, there are three promotional cards for upcoming races: '4 hours to LIVE race', 'Opening soon ...', and '9 days to LIVE race'. In the top right corner of the 'Open community races' section, there are two buttons: 'Manage races' and 'Create race'. The 'Create race' button is circled in red.

4. 「レースの詳細」ページで、[\[クラシックレース\]](#) 競技形式を選択します。これにより、設定した時間内にゲストが独自のスケジュールで参加できるようになります。あるいは [\[LIVE レース\]](#) で、リアルタイムイベントとして非公開または公開で放送できます。
5. コンペティション形式の選択に基づいて、[the section called “レースを作成する \(クイックスタート\)”](#) でクラシックレースの作成を続けるか、LIVE レースの作成を続ける場合は、手順 1~3 に従ってください。

6. お客様のレース日を選んだ後、レースのカスタマイズを展開します。



7. 競技トラックを選択します。人気: 最大から最小へ / 最小から最大へ、難易度: 最大から最小から最大へ、および長さ: 最長から最短へ / 最短から最長へで、トラックを並べ替えることができます。各カテゴリのトラックをすべて表示するには、[レーストラックのオプションをもっと見る方法]を選択します。展開されたメニューを閉じるには、[レーストラックオプションの数を減らす]を選択します。



8. 必要に応じて、参加者のイベントの目標とルールを要約したレースの説明を作成します。LIVE レースの場合は、イベントのビデオ会議またはライブストリームのリンクを追加します。説明はリーダーボードの詳細に表示されます。
9. クラシックレースの [ランキング方法] では、勝者が最速ラップを記録するレーサーならば [ベストラップタイム] から選びます。[平均時間] を選ぶと、イベントの時間枠内に複数回のトライア

ルの後、勝者は平均時間が最も短いレーサーになります。あるいは [合計時間] を選ぶと、勝者は総合平均が最も短いレーサーになります。すべての LIVE レースのリーダーボードの順位はベストラップタイムでランク付けされるため、このフィールドは表示されません。

10. クラシックレースの場合は、最小ラップの値を選択します。これはレースのリーダーボードに結果を提出するためにレーサーが完了しなければならない連続ラップ数になります。初心者のレースでは、小さめの数を選びます。上級ユーザーの場合は、大きめの数を使用します。このカスタマイズでは、デフォルトが 1 ラップであるため、LIVE レースでは利用できません。
11. 場外ペナルティを使用する場合、RL モデルがトラックから外れる時には、レーサーの時間に加算する秒数を選択します。
12. これで、タイムトライアルレースのすべてのカスタマイズオプションが完了しました。タイムトライアルレースフォーマットを選択した場合、[次へ] をクリックして、レースの詳細を確認します。[オブジェクトの回避](#)または [Head-to-bot](#) レース形式を選択した場合、適切な手順に進み、レースのカスタマイズを完了してください。
13. 「レースの詳細を確認する」ページで、レースの仕様を確認してください。設定を変更するには、「レースの詳細」ページで [編集] または [戻る] を選択します。招待リンクを取得する準備ができたなら、[送信] を選択します。
14. レースを共有するには、モーダルで [招待リンクのコピー] をコピーし、メール、テキストメッセージ、お気に入りのソーシャルメディア アプリケーションにペーストします。「<Your Race Name>」ページで、招待タブを選んでお客様のレースをシェアします。リンクはレースの終了日に期限切れになります。

The screenshot shows the AWS DeepRacer console interface. On the left is a navigation menu with categories like 'Racing League', 'Reinforcement learning', 'Resources', and 'Next challenge'. The main content area is titled 'TestLiveRace' and has a breadcrumb trail: 'AWS DeepRacer > Community races > Manage races > TestLiveRace'. There are two tabs: 'Race details' and 'Invitation', with 'Invitation' being the active tab. Below the tabs, there's a section for 'Invitation details' which includes a 'Share with race participants' section with a text input field containing a long URL and a 'Copy' button. Below that, there's a 'Suggested email template' section with a table of email content and another 'Copy' button.

15. [完了] を選択します。「レースを管理する」ページが表示されます。

メールテンプレートを使用して新しいレーサーを招待する方法、レースからレーサーを削除する方法、レーサーのモデル提出状況を確認する方法については、「[コミュニティレースの管理](#)」を参照してください。

オブジェクト回避レースのカスタマイズを完了するには

1. 衝突ペナルティを使用するには、オブジェクトまたはボットと衝突するレーサーの時間に加算される秒数を選択します。秒数が多いほど、チャレンジは大きくなります。

Collision penalty
Choose the number of seconds added to a racer's time for colliding with an object.

3

Number of objects.
Choose the number of objects a racer must avoid on the track.

4

Include random objects
Make the race more challenging by placing objects on the track.

Obstacle 1
Lane placement: Outside lane
Location (%) between start and finish: 20

Obstacle 2
Lane placement: Inside lane
Location (%) between start and finish: 40

Obstacle 3
Lane placement: Outside lane
Location (%) between start and finish: 60

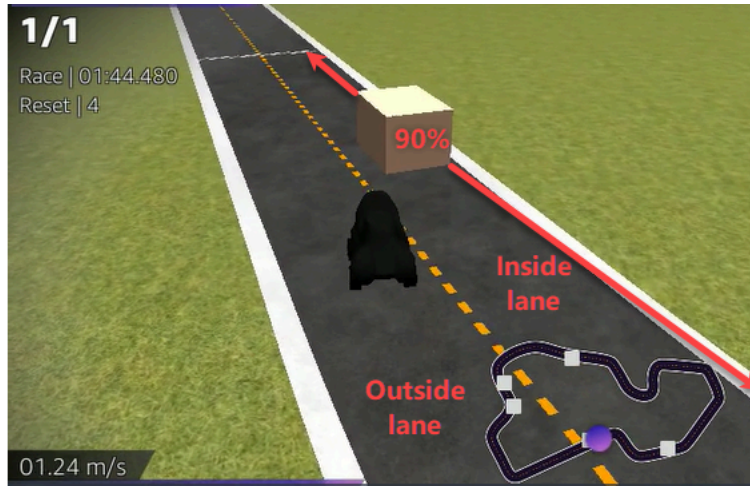
Obstacle 4
Lane placement: Outside lane
Location (%) between start and finish: 80

Community races visibility
Races are private. Only racers that are invited to a race can view it. To invite racers to your race, you share a link. Racers you've invited can forward the link to other racers. As the race organizer, you can revoke any racer's permission to race.

Cancel Next

2. オブジェクトの数を使用するには、レーサーがトラック上で回避しなければならない障害物の数を選択します。オブジェクトが多いほど、レースは難しくなります。

- レーストラックにランダムなオブジェクトを追加し、レーサーごとに異なる場所に配置するには、[ランダムオブジェクトを含める] を選択します。これは、レーストラック上の予期しないオブジェクトのようなランダムなイベントによく一般化するRLモデルを作成するためにより長い時間のトレーニングと報酬関数の試行錯誤が必要になるため、参加者にとってより困難になります。
- レーン番号またはレーンの配置のためのオブジェクトの位置を選択して、オブジェクトを配置する場所を選択します。トラックは中央線で半分に分割され、内側と外側の車線が作成されます。オブジェクトを内側または外側の車線のいずれかに配置できます。



- オブジェクトごとに、[開始から終了までの位置 (%)] の値を選択します。この数字は、オブジェクトを配置するトラックの開始ラインとフィニッシュラインの間の位置をパーセンテージで表しています。
- これで、オブジェクト障害レース特有の構成オプションが完了しました。[次へ] を選択します。
- 「レースの詳細を確認する」ページで、レースの仕様を確認してください。設定を変更するには、「レースの詳細」ページで [編集] または [戻る] を選択します。招待リンクを取得する準備ができたなら、[送信] を選択します。
- レースを共有するには、[招待リンクのコピー] を選択し、メール、テキストメッセージ、お気に入りのソーシャルメディアアプリケーションにペーストします。クラシックレースは全て非公開であり、招待リンクのあるレーサーのみ見ることができます。リンクはレースの終了日に期限切れになります。
- [完了] を選択します。「レースを管理する」ページが表示されます。

レースでできることの詳細については、[「コミュニティレースの管理」](#)を参照してください。

head-to-bot レースのカスタマイズを完了する方法

1. ボット車の数を使用する場合、参加者の AWS DeepRacer RL モデルと競合する車の数を選択します。ボット車はビデオゲームの AI 車両に似ています。それらは移動するランダムなオブジェクトであるため、静止したオブジェクトからの複雑さのステップアップです。トラック上のボットが多いほど、レースの難易度が上がります。6 つまで選択してください。


Number of bot cars
3
The number of bot cars must be between 1-6.

Bot car speed
4
The speed must be between 0.2-6 meters per second.

Enable lane change
Enable bot cars to change lanes.

Minimum lane change time
4
The minimum time between lane changes must be between 1-8 meters per second.

Maximum lane change time
4
The maximum time between lane changes must be between 1-8 meters per second.

 **Community races visibility**
Races are private. Only racers that are invited to a race can view it. To invite racers to your race, you share a link. Racers you've invited can forward the link to other racers. As the race organizer, you can revoke any racer's permission to race.

Cancel **Next**

2. ボット車の速度を使用する場合、ボット車がトラック内を移動する速度を選択します。速度はメートル/秒で測定されます。速度は 0.2 - 6 メートル/秒でなければなりません。
3. ボットがレーンの変更を許可し、レーサーの AWS DeepRacer RL モデルの課題をさらに複雑にしたい場合は、[レーン変更を有効にする] を選択して下さい。
4. 最小車線変更時間を使用する場合、ボット車がレーンを変更するインスタンス間を通過する最小秒数を選択します。
5. 最小車線変更時間を使用する場合、ボット車がレーンを変更するインスタンス間を通過する最小秒数を選択します。

- これで、head-to-bot race のすべてのカスタマイズオプションが完了しました。[次へ] を選択します。
- 「レースの詳細を確認する」ページで、レースの仕様を確認してください。設定を変更するには、「レースの詳細」ページで [編集] または [戻る] を選択します。招待リンクを取得する準備ができたなら、[送信] を選択します。
- レースを共有するには、[招待リンクのコピー] を選択し、メール、テキストメッセージ、お気に入りのソーシャルメディアアプリケーションにペーストします。クラシックレースは全て非公開であり、招待リンクのあるレーサーのみ見ることができます。リンクはレースの終了日に期限切れになります。
- [完了] を選択します。「レースを管理する」ページが表示されます。

レースでできることの詳細については、「[コミュニティレースの管理](#)」を参照してください。

LIVE の AWS DeepRacer コミュニティレースに参加するには

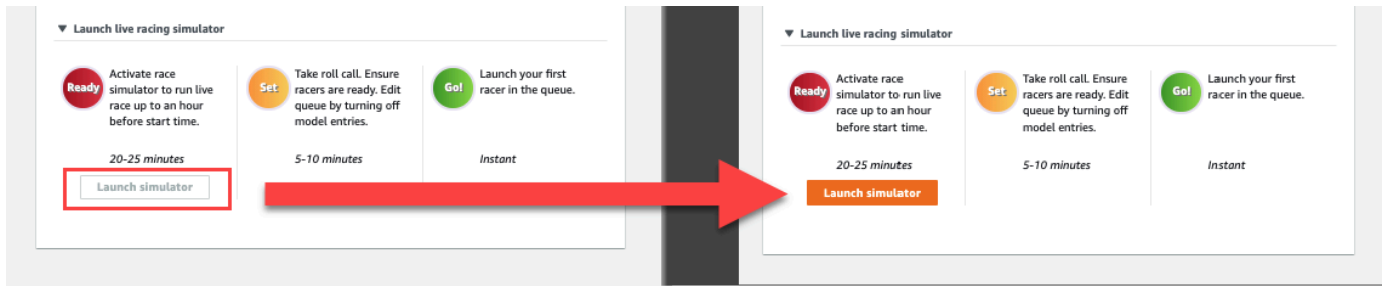
LIVE レースを作成してレーサーを招待しました。[the section called “LIVE レースを放送する”](#) からの支援を受けて、イベントの放送を非公開にするか、公開するかを決定しました。キューを管理し、レースシミュレータを設定し、レーサーを起動する方法を学びます。

開始する前に

- Chrome または Firefox のブラウザを使用してください (ブラウザが最新であることを確認してください)
- 仮想プライベートネットワーク (VPN) を使用している場合、VPN を切断してください
- 追加のタブをすべて閉じます

LIVE の仮想レースを実行するには

- 「コミュニティレース」ページで、モデレートしたいレースのレースカードを見つけ、レースを見るために [今すぐ参加] を選択します。
- [レース主催者コントロールパネル] の下にある [LIVE:<Your Race Name>] ページで、[シミュレータの起動] を選択します。このボタンは、レース開始時間の 1 時間前に使用可能になります。[レース主催者] コントロールパネルのこのセクションを非表示にするには、[LIVE レースシミュレーターを起動する] ヘッダーを選んでください。



3. [COMING UP] で、提出受け付けを終了するために [モデルエントリが開いている] をオフに切り替えて下さい。これにより、モデルの提出受け付けが終了し、トグルの下に編集可能なレーサーキューが作成されます。トグルがオフになるまで、レーサーを起動することはできません。

AWS DeepRacer > Community races > TestLiveRace > LIVE

LIVE: TESTLIVERACE View leaderboard

Start time: 2:00 PM local, July 2
 Time trial race
 Cumulo Turnpike track
 Best lap time
 Unlimited resets

LEADERBOARD

#1	---
#2	---
#3	---
#4	---
#5	---
#6	---
#7	---
#8	---

COMING UP

Model entries open
 Toggle off to edit race queue Edit

Racer up next Time

Race organizer control panel Open broadcast mode Declare winner!

Race simulator Refresh
 Status: Not created
Reset simulator

Current ranked submissions: 0
 Leaderboard can be cleared when no submissions are in progress.
Clear leaderboard ranking

Launch live racing simulator

Ready Activate race simulator to run live race up to an hour before start time. 20-25 minutes Launch simulator

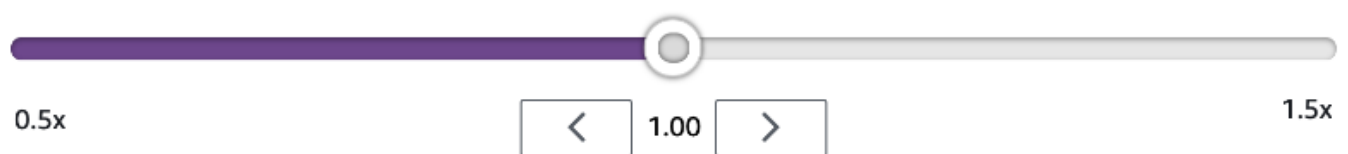
Set Take roll call. Ensure racers are ready. Edit queue by turning off model entries. 5-10 minutes

Go! Launch your first racer in the queue. Instant

4. 作成したビデオ会議を開き、レーサーを集めてください。
5. レーサーロールコールを開始します。
 - a. レーサーに確認して、はっきり聞こえるようにしましょう。

- b. まずビデオを使って自己紹介してください。帯域幅を最適化するために、後でシャットダウンすることをお勧めします。
 - c. 通話中の人のリストが、グループ内のレーサーのリストと一致していることを確認してください。
 6. モデルロールコールを開始します。
 - a. レーサーキュー内にあるエイリアスのリストがレーサーのエイリアスのリストと一致し、赤でハイライトされていないことを確認します。これは、モデルが正常に送信されなかったことを意味しています。
 - b. レーサーにチェックインして、モデルの提出に問題がないか確認してください。
 7. レースのスケジュールとルールを確認します。レーサーに自分の順番でレースをしなければならない時間を伝え、リーダーボードの順位はそのタイムフレーム内の最速ラップによって決まるということを思い出させてください。
 8. レース中にのみレーサーに表示される速度制御機能により、スピードコントロールスライダーを使用して手動で最高速度を設定できることを説明してください。これにより、モデルの速度パラメータは一時的に上書きされますが、ステアリング角度は上書きされません。モデルで引き続き操縦できますが、レーサーはレートを乗算して車の速度を増減するキーモーメントを選択できるようになりました。モデルの速度パラメータの使用に戻るには、レーサーは乗数を 1 にリセットします。スピードコントロールスライダーはアクセルペダルではなく、戦略的なリアルタイム調整の機会であることをレーサーに思い出させてください。

Speed control



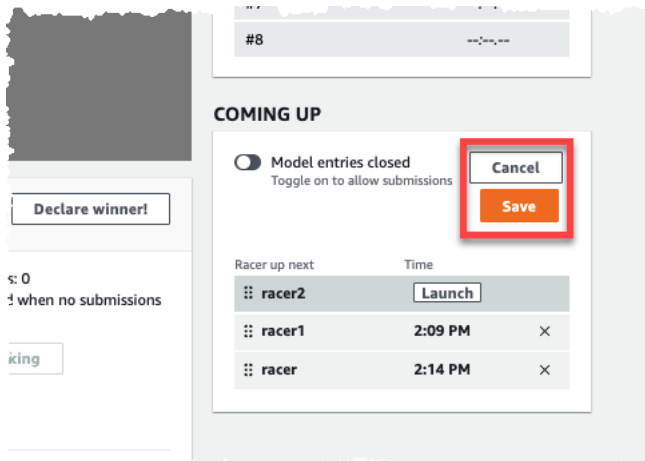
9. 次に、レースウィンドウのビデオオーバーレイには、レーサーのパフォーマンスを最適化するのに役立つ情報が含まれていることを説明します。トラックマップオーバーレイは、レーサーのペースに応じて色が変わる 3 つのセクターに分かれています。レーサーが個人ベストを記録したセクションは緑色で表示され、最も遅いセクターは黄色で表示され、セッションベストは紫色で表示されます。レーサーには、ベストラップタイム、残り時間 (m/s)、リセット、現在のラップタイムを詳述した統計情報も表示されます。



Track map overlay key:

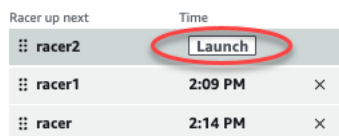
- Green - Personal best
- Yellow - Slowest sector
- Purple - Session best

10. レーサーの質問に答える。
11. オプションとして、[カミングアップ] で [編集] を選択してレーサー名を選択してドロップしてレースキューを並べ替えます。



13. キューの最初のレーサーを起動します。

- 手動で各レーサーを起動するには、トップレーサーキュー名の横に表示される [起動する] ボタンを押します。各レーサーのターンには、発進後、コンソールで「10、9、8、7、6...」のカウントダウンアニメーションが表示されます。
- スタートを切った後、リアルタイムで評価される間、モデルは選択した時間だけ実行されます。
- レース途中で故障したモデルの場合、レーサーの再起動が必要となる時には、[レーサー] キュー内のエイリアスの横にある [起動する] ボタンをクリックして再起動します。
- 現在のレーサーが終了する約 2 分前に、会議ブリッジを通してキュー内の次の 2 人のレーサーに連絡し、レースの準備が整っていることを確認します。
- 現在のレーサーが終了する 30 秒前に、次のレーサーに 30 秒の警告を出します。
- 現在のレーサーが終了したことを確認したら、すぐに次のレーサーを起動してください。レースの終了は、コンソールのチェッカーフラッグアイコンで示されます。レーサーのスピードコントロールが無効になり、ビデオ画面でレースのリプレイが開始されます。



- シミュレータで問題が発生している場合には、必要に応じて、[シミュレータのリセット] を選択します。
- または、何らかの理由でリーダーボードをリセットしたい場合は、[リーダーボードランキングをクリアする] を選択することもでき、すべてのエントリがクリアされます。

16. レース終了時に、[勝者を宣言する！] ボタンを押し、レーサーに最後の挨拶を行い、賞品の配布方法を説明し、質問に答え、ビデオ会議を終了します。

AWS DeepRacer League プロダクションプレイブックを使用して LIVE コミュニティレースをブロードキャストします。

LIVE レースは、指定された日時に発生するリアルタイムイベントです。これは、1つのプライベートビデオ会議を提供する小規模なイベントから、小規模なチーム、コメンテーター、Twitch などのような LIVE ストリーミングを使用する放送業者によって行われる大規模なイベントまでの広い範囲をカバーします。

主催者ロール

AWS DeepRacer LIVE イベント中に主催者がプレイできる役割を以下に示します。企画するイベントが複雑になればなるほど、参加に必要な支援も増えるでしょう。

主催者

レース主催者は、レースと関連するビデオ会議を設定し、レーサーを編成し、指導します。LIVE レース中、主催者は主催者コントロールを使用してキューを作成し、レーサーを起動し、勝者を表彰します。主催者は LIVE チャンネルに表示されません。

コメンテーター

コメンテーターは、開催中にレースについて話し合い、イベントのプレイバイプレイ、追加情報、イベントとその参加者の内部知識を提供します。コメンテーターは公開イベントのメインスピーカーです。

ブロードキャスト

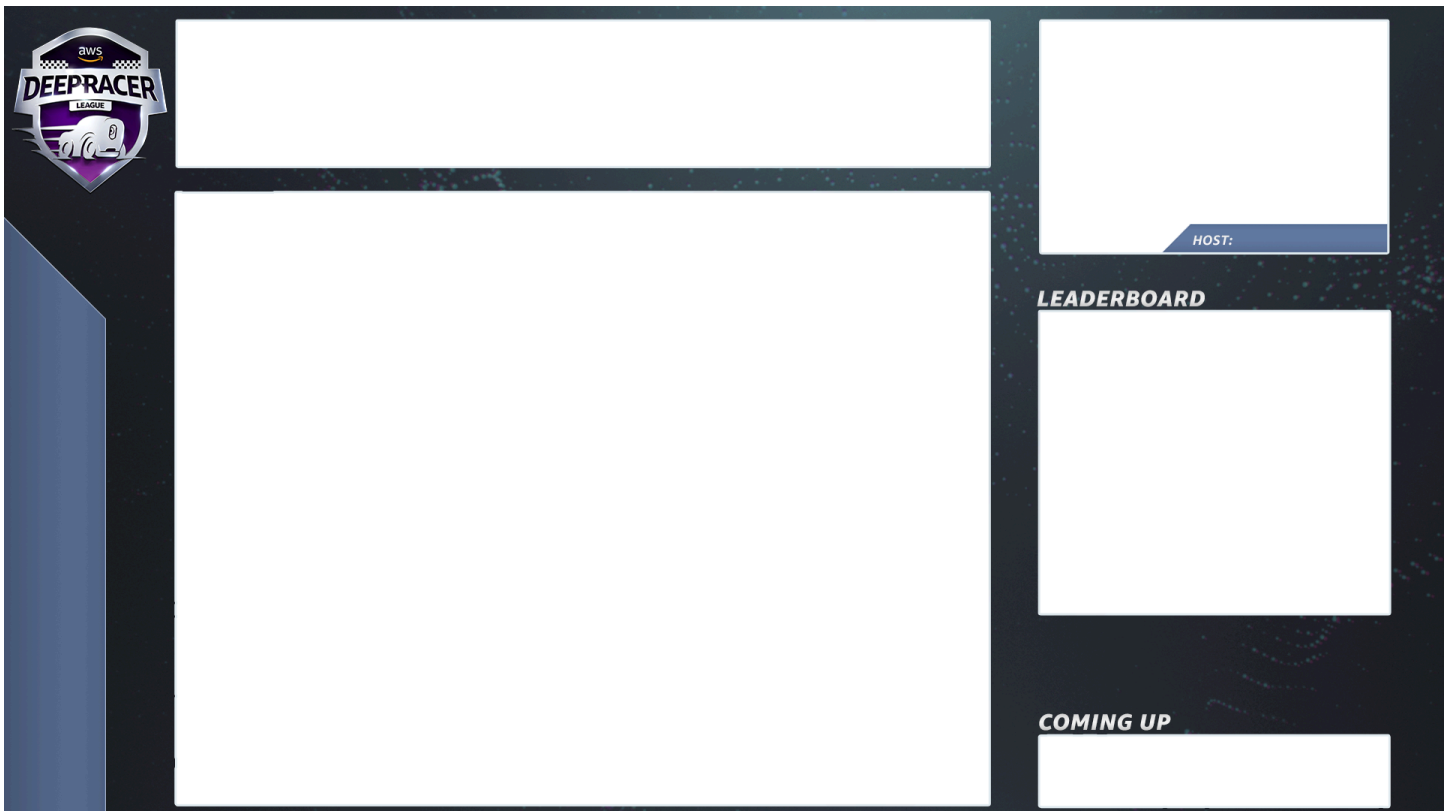
放送局はストリーミングソフトウェアを使用して事前にシーンを作成し、LIVE レース中にシーンを移行します。放送局はビデオフィールドも管理します。放送局は LIVE チャンネルに表示されません。イベント中はコンテンツのプロデューサーとして機能します。

ブロードキャスターのシーン

AWS DeepRacer イベントの LIVE ストリームでは、レースのストーリーが伝えられます。イベントの開始、中間、終了時にエンゲージメントを宣伝するには、シーンを使います。これらは、イベントのさまざまなセグメントを区切るグラフィックオーバーレイとビデオストリームで構成されるアニメーションとレイアウトです。

オーバーレイは、グラフィック（通常は透明 PNG ファイル）で、レースのブロードキャスターモードウィンドウと（オプション）ウェブカメラストリームまたはコメンテーターの上に置かれます。それはあなたのストリームのマスクのようなものです。コンテンツをその下に配置して、すべてをシームレスに並べて 1 つの統一されたレイアウトを作成します。

OBS などのストリーミングソフトウェアを使用して、ブロードキャスト前にシーンを設定します。イベント中にスムーズに移行し、ダイナミックなペースと観客の喜びを作り出します。たとえば、イントロアニメーションシーンを使用してイベントをキックオフします。次に、プライマリコンテンツシーン (PCS) に移行します。PCS は、レースビューとコメンテーター用の 1 つまたは 2 つのウィンドウを含むメインレイアウトです。フルスクリーンのデュアルコメンテーター、またはコメンテーターとインタビュー対象者のシーンにカットして、物事を生き生きとした状態に保ち、リーダーボードのシーンで終わります。オプションで、レースの間にカットインする商業シーンを作成します。



AWS DeepRacer シーンテンプレート

AWS DeepRacer League 仮想サーキットチームは、LIVE コミュニティレースに使用するテンプレートファイルのコレクションを作成しました。[AWS DeepRacer シーンテンプレート](#)をダウンロードし、それらを使ってプロ並みのイベントを放送します。

シーンタイプとその使用方法

1. AWS DeepRacer シールドアニメーションのイントロ

2. コンソール共有のみ表示:

- ベースレイヤー-レースのブロードキャスターモード URL のスクリーンシェア。シーンのフレームに合わせてサイズを変更します。

3. シングルコメンテータービュー (1up):

- ベースレイヤー - レースのブロードキャスターモード URL のスクリーンシェア。シーンのフレームに合わせてサイズを変更します。
- 次のレイヤー- 撮影しているコメンテーターが同じ部屋にいる場合は、OBS Ninja またはローカルウェブカメラ。ピクチャ内の右上の画像 (PIP) ウィンドウのシーンフレームの下にプルインしてサイズを変更します。

4. コメンテーター + インタビュー対象者またはデュアルコメンテーター (2up):

- ベースレイヤー - レースのブロードキャスターモード URL のスクリーンシェア。シーンのフレームに合わせてサイズを変更します。
- 次のレイヤー- 撮影しているコメンテーターが同じ部屋にいる場合は、OBS Ninja またはローカルウェブカメラ。ピクチャ内の右上の画像 (PIP) ウィンドウのシーンフレームの下にプルインしてサイズを変更します。
- デュアルウェブカメラフィールドまたは ninja フィールドを右上のウィンドウに収まるようにサイズ変更します (イベントの1週間前にセットアップ中-AVはすべてのフィールドをチェックし、OBSでカメラを割り当てます)

5. デュアルコメンテーター全画面表示 (レースビューなし、インタビューのみ):

- ベースレイヤーコンソールはありません。カメラフィールドは 2 つだけです。

6. リーダーボードの終了

- リアルタイムで、シーンレイヤーの上にリーダーボードの結果を手動で入力します。

AWS DeepRacer シーンテンプレートヒント

- 34 - コメンテーター用のタイトルを設定します (Pips で名前のシーンを事前に構築します)
- 234 - レースビュー
 - 左上の AWS DeepRacer League ロゴを会社のロゴに置き換えることを検討してください。
 - 左下のテキストをレース名に置き換え、縦書きの情報を入れ替えます。

LIVE プライベートブロードキャストを配信するには - 5 分セットアップ

AWS DeepRacer LIVE コミュニティレースの非公開の放送は、小規模で非公式のレースに適しています。

主催者ロール

- スタンダードレースの場合、主催者は 1 人だけ必要です。

ハードウェア

- 推奨ハードウェア - 最低 16 GB の RAM
- (オプション) 高品質マイク、ヘッドセット、または AirPods
- (オプション) LED リングライト - 眼鏡にリングライトが反射するのを防ぐには、着用者の顔に対して角度を付けてください。
- (オプション) ウェブカメラと GoPro - 映像を多様化するため

ヒント

- Chrome または Firefox のブラウザを使用します (ブラウザが最新であることを確認してください)
- 使用している場合、VPN から切断します。
- 追加のタブをすべて閉じます

非公開の LIVE AWS DeepRacer イベントを実行するには

1. [\[AWS DeepRacer コンソール\]](#) を開きます。
2. [\[コミュニティレース\]](#) を選択します。
3. 「コミュニティレース」ページで [\[レースを作成\]](#) を選択します。
4. 標準の LIVE コミュニティレースを主催する日時を決めます。
5. LIVE コミュニティレースを作成する手順を実行する前に、[\[レース日\]](#) でこの時間枠が使用可能であることを確認してください。LIVE コミュニティのレースは 4 時間でもかまいません。より長いレースをスケジュールするには、カスタマーサポートにお問い合わせください。
6. 利用可能な日時を決めたら、レースの主催者と参加者のための対応するビデオ会議を作成します。観客がほとんどいない小規模なレースを走っている場合は、1 つのビデオ会議だけで済みます。大規模なプライベートレースを走らせたい場合は、レースを視聴者に放送するための別のビデオ会議を作成してください。

7. [the section called “レースを作成する \(クイックスタート\)”](#) の手順を行います。次に [LIVE レースの作成を続けるには] を選択します。
 - オプションとして、ステップ 8 で、メールテンプレートの推奨の隣にある [コピー] を選択し、次にレーサーやレース主催者向けのメールを作成します。コピー推奨メールテンプレートの横に、賞品、モデル提出時間枠、およびレーサーが出会ってレースの準備をする会議ブリッジのリンクを記入します。
8. レース当日は [the section called “LIVE レースで走行する”](#) の指示に従ってください。
9. 賞品があれば、レース参加者に配布します。

LIVE の公開放送を制作するには - 2 時間設定

AWS DeepRacer LIVE コミュニティレースのプレミアムブロードキャストは、複数のブロードキャストシーンを使用します。3 人以上のスタッフが、グローバルストリーミングプラットフォームでレースを放送します。以下の手順は、Twitch を例として使用します。

主催者ロール

- 主催者
- コメンテーター/MC
- ブロードキャスト
- Twitch モデレーター-オプション

ハードウェア

- 推奨ハードウェア:16 GB 以上の RAM が必要です。
- (オプション) 高品質マイク、ヘッドセット、または AirPods
- (オプション) LED リングライト - 眼鏡にリングライトが反射するのを防ぐには、着用者の顔に対して角度を付けてください。
- (オプション) ウェブカメラと GoPro - 映像を多様化するため

ヒント

- Chrome または Firefox のブラウザを使用してください (ブラウザが最新であることを確認してください)
- VPN を使用している場合は、VPN を切断します。

- 追加のタブをすべて閉じます

前提条件

- [Twitch アカウント](#) - LIVE ビデオストリーミングサービス。
- Twitch ストリームキー - 動画の送信先をソフトウェアに知らせます。
- [オープンブロードキャスタソフトウェア \(OBS\)](#) - ビデオ録画と LIVE ストリーミング用のオープンソースの無料ソフトウェア。
- (オプション) [VDO Ninja \(旧称、OBS Ninja\)](#) - コメンテーターとインタビュアーを含める場合、追加のビデオフィードを追加して切り替えるためのツール。

プライベート LIVE の AWS DeepRacer イベントを実行するには

1. [Twitch アカウントにサインアップする方法](#)の手順に従って、[Twitch](#) アカウントをセットアップします。
2. Twitch ストリームキーを見つけます。[Twitch Steam キー](#)を検索する方法について説明します。
3. [\[オープンブロードキャスタソフトウェア \(OBS\)\]](#) をダウンロードします。
4. [OBS](#)の使用方法について学習します。事前に設定してください。レースの 1 週間前までにアセットを準備することをお勧めします。
 - a. 付属する AWS DeepRacer シーンテンプレートをダウンロードします。
 - b. シーンをロードして修正します。
 - c. レース URL でソースを更新します。
 - d. カメラをチェックしてください。
 - e. ユーザーをフィードに割り当てます。
5. オプションで、コメンテーターとインタビュー対象者がブロードキャストイベントに参加している場合は、[VDO Ninja \(旧称、OBS Ninja\)](#) を使って複数のビデオフィードを管理できます。[Ninja](#) の使用方法について学習します。
6. [AWS DeepRacer コンソール](#) を操作して、レースを作成しましょう。
7. [コミュニティレース] を選択します。
8. [コミュニティレース] で、[レースを作成] を選択します。
9. パブリック LIVE コミュニティレースを主催する日時を決めます。
10. LIVE コミュニティレースを作成する手順を実行する前に、[レース日] で、この時間枠が使用可能であることを確認してください。LIVE レースのデフォルト期間は 4 時間です。より長いレー

スをスケジュールするには、カスタマーサポートにお問い合わせください。LIVE レースが4時間未満の場合、実行するアクションはありません。

11. 利用可能な日時を決めたら、レースの主催者と参加者のための対応するビデオ会議を作成します。
12. 次に、放送局用に別のビデオ会議を作成します。
13. LIVE コミュニティレースを設定する手順を行います。
 - a. 必要に応じて、ステップ 8 の [レースの説明] で、レーサーが家族や友人と共有するための LIVE ストリームのリンクを追加します。レーサー用のレーサールームカンファレンスブリッジを含めることもできます。説明はリーダーボードの詳細に表示されます。
 - b. オプションとして、ステップ 12 で、[メールテンプレートの推奨] の隣にある [コピー] を選択し、次にレーサーやレース主催者向けのメールを作成します。賞品、モデル提出の時間枠、およびレーサーが集合してレースの準備をする会議ブリッジのリンクを記入します。
 - c. 主催者のチーム用に別のメールやチャットを作成します。
14. レース当日は、[the section called “LIVE レースで走行する”](#) の指示に従ってください。
15. 勝者と参加者を表彰し、賞品を配布し、ブログを書き、つぶやき、投稿、拡散します。

AWS DeepRacer コミュニティレースを管理する

すべてのコミュニティレースは非公開です。これらは、招待リンクを受け取った個人にのみ表示されます。参加者は招待リンクを自由に転送できます。ただし、レースに参加するには、参加者に AWS アカウントが必要です。初めてのユーザーは、レースに参加する前にアカウントの作成プロセスを完了させる必要があります。

レース主催者は、開始日と終了日などのレースの詳細を編集したり、参加者を削除したりできます。

AWS DeepRacer コミュニティレースに参加する方法

1. AWS DeepRacer コンソールにサインインします。
2. [コミュニティレース] を選択する。
3. 「レースを管理する」ページで、レースのために自分が管理するレースを選択します。参加者のリストなど、選択したレースの詳細が表示されます。

AWS DeepRacer > Community races > Manage races

Manage races

Races (9) Actions ▾ Create race

< 1 >

	Name ▾	Status ▾
<input type="radio"/>	TESTSrFUN	Open
<input checked="" type="radio"/>	MyRaceName	Open
<input type="radio"/>	Fun2LearnRL	Opening soon
<input type="radio"/>	adfdas	Opening soon
<input type="radio"/>	Supa Awesome Fast Race	Opening soon
<input type="radio"/>	Need4Speed	Opening soon
<input type="radio"/>	BestRaceEver	Closed
<input type="radio"/>	AnotherRace	Closed
<input type="radio"/>	EasyRace	Closed

MyRaceName Copy invitation link

Status	Race dates (GMT)	Race track
Open	09/30/2020 - 10/01/2020	Cumulo Turnpike

Racers (0) Remove racer

< 1 >

	Alias ▾	Date joined ▲
--	---------	---------------

- レースの詳細を編集するには、[アクション]で、[レースの詳細の編集]を選択します。

AWS DeepRacer > Community races > Manage races

Manage races

Races (9) Actions ▲ Create race

Search races

Name	Status
<input type="radio"/> TESTSrFUN	Open
<input checked="" type="radio"/> MyRaceName	Open
<input type="radio"/> Fun2LearnRL	Opening soon
<input type="radio"/> adfdas	Opening soon
<input type="radio"/> Supa Awesome Fast Race	Opening soon
<input type="radio"/> Need4Speed	Opening soon
<input type="radio"/> BestRaceEver	Closed
<input type="radio"/> AnotherRace	Closed
<input type="radio"/> EasyRace	Closed

MyRaceName Copy invitation link

Status	Race dates (GMT)	Race track
Open	09/30/2020 - 10/01/2020	Cumulo Turnpike

Racers (0) Remove racer

Search racers

< 1 > ⚙️

Alias	Date joined
-------	-------------

画面の指示に従って編集を完了します。

5. イベントのリーダーボードを表示するには、[アクション] で、[リーダーボード] を選択します。

6. イベントの招待リンクをリセットするには、[アクション]で、[招待のリセット]を選択します。招待リンクをリセットすると、前のリンクを選択してもレースにアクセスできなくなります。すでにリンクをクリックしてモデルを提出したすべてのユーザーは、レースに残ります。

必要に応じて、このステップでリンクをコピーして招待された参加者と共有できます。

7. オープンレースを終了するには、[アクション]で、[レースを閉じる]を選択します。このアクションは、指定された終了日前に直ちにレースを終了します。
8. イベントを削除するには、アクション から [レースを削除] を選択します。このアクションは、すべての参加者のコミュニティレースから、このレースと詳細を完全に削除します。
9. 参加者を削除するには、1人以上のレース参加者を選択し、[参加者を削除]を選択して、参加者の削除を確認します

イベントから参加者を削除すると、レースイベントへのアクセス許可が取り消されます。

AWS DeepRacer イベントの主催

AWS DeepRacer イベントとは何ですか？

AWS DeepRacer は、人工知能や機械学習 (AI/ML) を実践的に学習するための興味深い方法を提供する教育サービスです。AWS DeepRacer は、AI/ML 分野の人材不足を解消し、ビジネスニーズに AI/ML を適用する組織を支援します。

AWS DeepRacer では、チームに AI/ML スキルを導入できるだけでなく、チームビルディングや友好的な競争を促すイベントを主催することもできます。これらのイベントでは、AWS DeepRacer League で対面または仮想環境で AWS DeepRacer カーでレースする強化学習モデルの作成実習を提供します。これは、技術者と非技術者の両方の参加者が機械学習の基礎を学ぶのに役立ちます。AWS DeepRacer イベントは、リーダーがチームと連携して組織の AI/ML のビジョンと目標を達成するのに役立ちます。

このガイドでは、独自の仮想または対面による AWS DeepRacer イベントの計画と開催を開始するために必要なリソース、ツール、例を紹介しています。参加者が 50 人以下で AWS DeepRacer イベントを計画する場合は、「[始める前に考慮すべきこと](#)」を参照してください。大規模なイベント (参加者 50 人以上) を計画している場合は、AWS アカウントチームと協力して[イベントをリクエスト](#)することをお勧めします。

AWS DeepRacer イベントの利点に関する詳細とお客様の声を確認するには、「[AWS DeepRacer インタープライズイベント](#)」を参照してください。

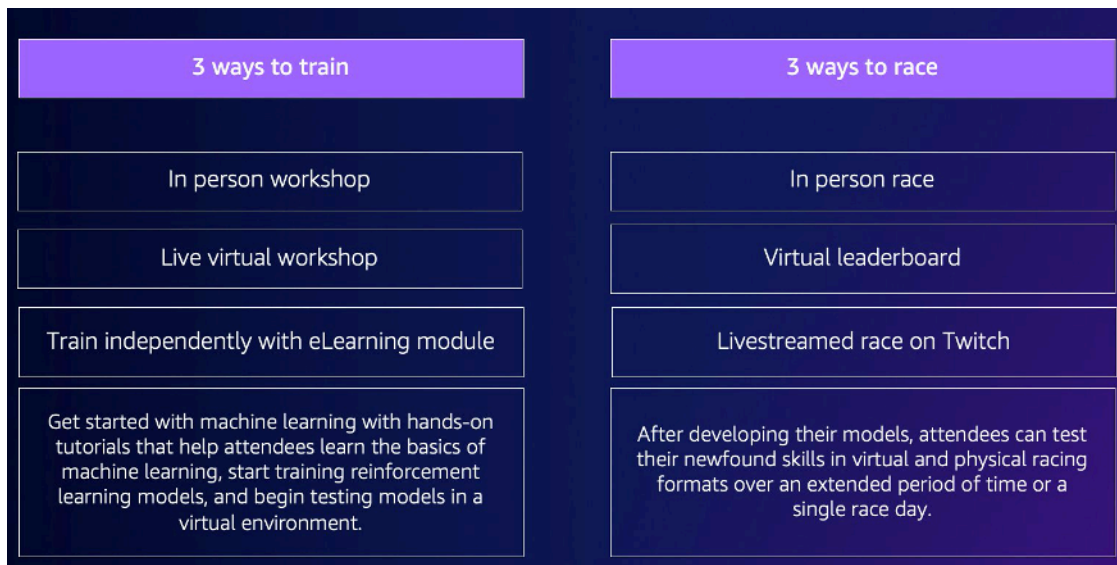
AWS DeepRacer イベントの仕組みと期待する内容

チームに強化学習教育と実践を提供したい場合、組織に新しい人材を呼び込むのを促進したい場合、あるいはその両方を提供したい場合でも、このガイドでは、AWS DeepRacer の独自の活動を作成してカスタマイズするのに役立つツールとリソースを提供しています。

AWS DeepRacer イベントはニーズや目的に合わせて柔軟に対応できますが、AWS DeepRacer イベントの基本的な公式には以下の項目が含まれます。

- 「[オンラインの AWS DeepRacer 強化学習コース](#)」や、AWS DeepRacer Pit Crew の専門家による対面ワークショップなどの教育コンポーネント。
- AWS コンソールでのハンズオンモデルトレーニングと、参加者がモデルを作成してトレーニングする時間。

- AWS DeepRacer レースでは、単一の対面レースや複数の仮想レースなど、参加者がトレーニングした機械学習モデルを実際に見ることができます。主催できるレースの種類の詳細については、「[AWS DeepRacer レースの種類](#)」を参照してください。
- イベント参加者向けのイベント後の要約または次のステップに関するコミュニケーション。



AWS DeepRacer イベントに期待される内容や顧客のハイライトについては、「[AWS DeepRacer Accelerate の YouTube 動画](#)」をご覧ください。

始める前に考慮すべきこと

イベントを計画する際の最初のステップでは、組織のビジネス目標とゴールを定義してから、プロジェクト計画を立てます。目標の例としては、「自分の組織で楽しく教育的な方法でチームビルディングを奨励したい」などがあります。

この計画で、以下の質問に答えるようにします。

- 「イベントリーダーは誰ですか？」

組織の誰がこのイベントの計画と実施を主導しているのかを特定してください。イベントリーダーには、施設、人事、PR/マーケティング、エグゼクティブスポンサー、AWS アカウントチームなどのチームが含まれる場合があります。

- 「イベントの責任者は誰ですか？」

組織内でイベントを担当している人物を特定します。このイベントを計画する際、この人物またはチームが主な主催者となり、すべての決定に参加する必要があります。

- イベントの開始日は？

イベントをいつ開催するか決めてください。大規模なイベント (参加者が 100 名以上) をお考えの場合は、イベントの計画や進行に十分なリードタイムを確保するため、開催日をさらに先に設定する必要があります。タイムラインの例については、「[AWS DeepRacer イベントの例](#)」を参照してください。

- 推定予算はいくらですか？

イベントの予算を見積もってください。イベントの費用に関する考慮事項は 2 つあります。

- イベントの費用: 対面式のイベントを主催する場合、これにはイベントの場所、トラック、デバイスの購入から、食べ物、飲み物、イベントの賞品まで、あらゆるものが含まれます。

- モデルトレーニングコスト: このコストは、AWS DeepRacer サービス AWS マネジメントコンソールを使用してで強化学習モデルをトレーニングする従業員が負担します。たとえば、各参加者が AWS DeepRacer コンソールで 1 つ以上のモデルをトレーニングするには、約 10 時間かかります。費用の詳細については、「[AWS DeepRacer の価格設定](#)」を参照してください。

- イベントにはどのような場所が必要ですか？

対面式のイベントを開催する場合は、ワークショップやレースのための物理的な場所を確保する必要があります。選択する物理的なトラックのタイプもその場所に収まる必要があります。

- 参加者の推定人数はどのくらいですか？

イベントに参加する参加者の数を見積もってください。参加者の数も推定予算に影響します。

- 成功度を測るために、イベント終了後に回顧セッションを開く予定はありますか？

参加者に ML スキルの向上とコラボレーションの継続を促すため、次のステップを伝え、参加者にフィードバックを求めることを検討してください。例えば、参加者にアンケートを送って関心度を評価し、組織内の AI/ML リーダーを募集します。これにより、将来の AWS DeepRacer イベントに誰が関与するか決定される可能性もあります。

- 組織内でイベントを推進するためのコミュニケーション戦略は何ですか？

このイベントは、社内のメールやカレンダーによる招待状と同じように、組織内で簡単に伝えることができます。

- 経営陣のステークホルダーは誰ですか？

イベントのスポンサーとして、組織内での思想的リーダーシップとコラボレーションを奨励するエグゼクティブを特定します。

AWS DeepRacer レースの種類

プロジェクト計画の主な質問に答えた後、イベントでどのようなレースをするか決めます。開催するレースの種類は、推定予算とイベントの参加者数に基づいて決定する必要があります。次の2種類のレースを開催できます。

仮想レース

仮想レースは、複数のリージョンや遠隔地のチームが協力して、トレーニングを受けたモデルを仮想環境でレースさせるための優れた方法です。仮想レースは物理的なデバイスやトラックを必要としないため、グループの規模を柔軟に調整でき、場所や予算に制約のある組織に最適なソリューションです。

仮想レースには、クラシックレースとライブレースの2つのタイプがあります。クラシックレースとライブレースの主な違いは、クラシックレースの参加者が自分の好きな時間にモデルをトレーニングして提出できることです。ライブレースでは、参加者は設定された時間内に次々とモデルでレースします。どちらのタイプのレースも非公開または公開で共有でき、リーダーボードを使用して参加者のランキングを決定します。AWSでは、Twitchでレースを放送するためのツールも提供しています。

AWS DeepRacer レースの使用を開始するには、仮想レースが最も迅速な手段です。参加者は、機械学習や強化学習の紹介を含む入門プロセスをコンソールで進めることができます。コンソールでは、参加者が最初のモデルを作成する手順を案内します。仮想レースの設定方法の詳細については、「[レースの作成](#)」を参照してください。

対面レース

対面式のレースを開催して、チームが直接参加できる魅力的なイベントを開催しましょう。対面式レーシングには、物理的なトラックとAWS DeepRacer 車両デバイスが含まれます。AWSでは、イベント用のトラック構築と車両調達のオプションを提供しています。

AWS DeepRacer デバイスの詳細については、「[AWS DeepRacer 車両の運用](#)」を参照してください。物理的なトラックと障壁の構築および注文の詳細については、「[物理的なトラックを構築する](#)」を参照してください。

この2種類のレースに加え、教育的なオンライントレーニングやワークショップを実施したり、Twitchでレースイベントをライブでストリーミングしたりすることもできます。

ベストプラクティス

AWS DeepRacer イベントの計画をさらに効率的にするために、以下のベストプラクティスもお勧めします。

- 社内のリーダーのサポートを受けましょう。組織のリーダーからのサポートは、チームの参加を促し、エンゲージメントと参加者全体の満足度を向上させます。
- AWS アカウントマネージャーがいる場合は、担当者に連絡してイベントについて通知し、サポートが必要かどうかを話し合います。
- 大規模なイベント (50 人を超える参加者) を計画している場合は、早期に頻繁に AWS アカウントマネージャーとやり取りしてください。AWS アカウントチームの誰に連絡すればよいかわからない場合は、リクエストを提出してください。リクエストを提出するには、「[イベントをリクエストする](#)」を参照してください。イベントやトレーニングに関する AWS DeepRacer コミュニティからのその他のリソースについては、「[AWS DeepRacer Slack チャンネル](#)」をご覧ください。
- AWS DeepRacer イベントの目標を満たす予算を設定してください。予算は計画するイベントの種類に影響する可能性がありますが、AWS DeepRacer には、コストの制限と管理に役立つ「[マルチユーザーモード](#)」などのツールが用意されています。
- 参加者に賞品を用意します。チームが従来のチャンピオンシップカップや NFT に関心があるかどうかにかかわらず、参加を促す魅力的な賞品を用意しましょう。

AWS DeepRacer イベント入門

組織の目標が決まったら、プロジェクト計画を使用して開催したいイベントのタイプを絞り始めることができます。以下の目標例は、要件と AWS DeepRacer から得たいメリットに基づいてイベントを設定する方法を示しています。

チームビルディング

小規模なグループのチームビルディングを促進する 1 回限りのローカルイベントを開催したい場合は、対面イベントまたは仮想イベントを検討してください。この目標を達成するイベントの種類については、「[仮想イベントの例](#)」を参照してください。

AI/ML 教育への投資

技術系や非技術系の従業員に機械学習に慣れ親しんでもらい、これらのスキルを応用してもらいたいのであれば、より多くのリーダーシップの支援を求め、より多くの参加者が関与するチーム横断的なイベントにすることで、より大きな組織的なインパクトを生み出すことを検討する必要があります。

このイベントには、参加者を増やすことができるように、対面イベントと仮想イベントの両方の要素を含めることができます。この目標を達成するイベントの種類については、「[対面式イベントの例](#)」を参照してください。

組織内の認知度の促進と創出

組織を革新的で思想的リーダーとして位置づけ、より多くの人材を惹きつけ、組織内の一般的な認知度を高めたいのであれば、仮想または対面式イベントのライブストリーミング、またはカスタムイベントの作成を検討してください。この目標を達成するイベントの種類については、「[カスタムイベントの例](#)」を参照してください。

AWS DeepRacer イベントの例

以下のセクションでは、目標とプロジェクト計画の要件に基づいて作成できるさまざまなタイプのイベントの例をいくつか示します。これらのイベントのスケジュールは、教育とレースの要素に基づいてスケジュールされています。ただし、組織のニーズに応じて、イベントのタイムラインをいくつでもイベントのコンポーネントに合わせてカスタマイズできます。

仮想イベントの例

仮想イベントは、複数の場所にまたがる組織や複数の地域のチームが便利で費用対効果の高い方法で集まるための優れた方法です。仮想イベントは依存関係が少ないため、便利で費用対効果も高くなります。たとえば、対面式イベントのように物理的なトラック、デバイス、場所のスペースを考慮する必要はありません。以下に示す仮想イベントの例は、タイムラインや参加者数など、さまざまなプロジェクト計画の項目に焦点を当てています。

	Virtual classic	Virtual live	Leaderboard + LIVE
Lead time	4 weeks	4 weeks	6-8 weeks
Cost	Organizing and submitting models in any virtual race is free. Costs incurred include: training a model, evaluating a model, and S3 storage of models.		
Recommended number of attendees	10-1000	<30	10-1000

• 仮想クラシックレース 2 週間のイベント:

以下に示す 2 週間のイベントスケジュールの例は、クラシックレースの提出に制限がなく、ライブレースの時間的制約もないため、多くの参加者やチームが参加を予定しているイベントを簡単に

開催したい組織には最適のオプションです。すべての参加者は、レース期間内であればいつでも、あるいは同時にモデルを提出できます。このイベント例では、AWS コンソールのコミュニティレースセクションを使用してプライベートクラシックレースを作成し、トレーニングとレースを2週間でスケジュールします。コミュニティレースの設定方法については、「[レースの管理](#)」を参照してください。

- 1 日目: 誰でも参加できる無料のオンデマンド仮想ワークショップに参加します。オンラインワークショップリソースの詳細については、「[AWS DeepRacer: 強化学習による推進](#)」を参照してください。また、AWS DeepRacer Pit Crew の専門家が仮想ワークショップを開催するようにスケジュールを立てることもできます。
- 1 ~ 5 日目: 参加者は AWS DeepRacer コンソールでモデルのトレーニング、更新、テストを実施します。参加者はモデルを提出し、1 日目 ~ 5 日目にかけて非公開の仮想レースに出場します。1 つの AWS アカウントで複数の参加者に AWS コンソールを設定する方法については、「[マルチユーザーモード](#)」を参照してください。
- 5 日目 ~ 10 日目: 5 日目に上位 10 名の勝者が決定され、翌週に新しいトラックで開催される非公開の新規仮想レースに参加できるようになります。
- 10 日目: 上位 3 人の勝者は 10 日目に決定され、レースとイベントは終了します。
- 「仮想ライブレースの 1 週間にわたるイベント:」

以下に示す仮想ライブレース イベントの例は、小規模チームにリアルタイムでレースの興奮と楽しさをもたらす素晴らしい方法です。この種のイベントは予算が低く、誰もがリアルタイムでレースに参加することができます。LIVE レースの運営方法について詳しくは、「[ライブレースの実行](#)」を参照してください。

- 1 日目: 参加者は AWS DeepRacer Pit Crew の専門家による仮想ワークショップに参加します。ワークショップの詳細については、「[AWS DeepRacer イベント](#)」を参照してください。
- 1 ~ 5 日目: 参加者は AWS DeepRacer コンソールでモデルのトレーニング、更新、1 日目 ~ 5 日目までのテストを実施します。
- 5 日目: 参加者は 1 ~ 2 時間オンラインで集まってモデルを提出し、5 日目のライブ仮想レースに参加します。
- 「仮想のクラシックレースとライブレースのイベント:」

以下に示すイベント例は、複数のリージョンにわたる多数の参加者、またはオンラインで複数のオフィスにまたがってリモートで作業する参加者を集めたい組織にとって最適なオプションです。この種のイベントでは、イベントの期間が2週間にまたがるため、参加者はモデルやレースのトレーニングを練習する機会が増えます。また、ライブ配信を視聴している参加者がイベントを盛り上げることができるように、ライブレースのイベントにはアナウンサーを配置することをお勧めし

ます。「[プロ部門フィナーレの Twitch ストリーム](#)」をチェックし、フィナーレ放送局がどのようにレースを盛り上げるか見てみましょう。

- 1 日目: 参加者は AWS DeepRacer Pit Crew の専門家による仮想ワークショップに参加します。ワークショップの詳細については、「[AWS DeepRacer イベント](#)」を参照してください。
- 1 ~ 5 日目: 参加者は AWS DeepRacer コンソールでモデルのトレーニング、更新、テストを実施してから、1 日目 ~ 5 日目までの非公開のリーグレースに参加するモデルを提出します。
- 5 日目: 上位 10 名の勝者が決定され、翌週に新しいトラックで開催される非公開の新規リーグレースに参加できるようになります。
- 5 日目 ~ 10 日目: 翌週、上位 10 名の勝者が決定され、Twitch でイベントをライブストリーミングしながらレースを行います。詳しくは、「[LIVE レースの放送](#)」を参照してください。

対面イベントの例

参加者を 1 つの物理的なスペースに集めることができる組織にとって、AWS DeepRacer を体験するには、対面式イベントに勝る方法はありません。トラックの脇に立って、実際の車でモデルがパフォーマンスを発揮するのを見るスリルと興奮を体験するのに勝るものはありません。通常、対面式のイベントは仮想イベントよりも多くのリソースを必要とし、費用もかかります。50 名以上の参加者と必要な予算があれば、AWS DeepRacer を対面で体験することを強くお勧めします。また、コースの脇で観戦している参加者にリアルタイムで競技結果が発表されるので、レースをよりエキサイティングなものにするため、対面式イベントにはアナウンサーを配置することをお勧めします。

	One-day workshop + race	Two-day workshop + race	Custom format
Lead time	6+ weeks	6+ weeks	10+ weeks
Recommended number of attendees	51-100	100-500	500-1,000

- 「対面式の 1 日限りのイベント:」

次の対面イベントの例は、ローカルチームにとって最適であり、すべての参加者が最初のモデルをトレーニング AWS マネジメントコンソールし、1 日で競争できるようにします。一般的な 1 日の対面式イベントは、参加者全員を対象にした対面式のワークショップとレースから成ります。通常、ワークショップは AWS DeepRacer Pit Crew の専門家が主導し、強化学習の基礎を学び、参加者に最初のモデルをトレーニングする機会を提供します。ワークショップの後、参加者はモデ

ルを提出して AWS DeepRacer デバイスにアップロードし、同じ日にレースを行うことができます。

- 参加者は、AWS DeepRacer Pit Crew の専門家による対面式の AWS DeepRacer ワークショップ (90 ~ 120 分) に参加します。AWS DeepRacer ワークショップをリクエストする方法の詳細については、「[イベントをリクエストする](#)」を参照してください。
- ワークショップ終了後の対面式レース (1 回のレースにつき 120 分、または最大 5 分)。
- 対面式の 2 日間のイベント:

次の 2 日間の対面式イベントの例は、ワークショップと実際のレースを別の日に分散させることで、参加者がレースに向けてトレーニングやモデルのアップデートを行う時間をより多く確保できることを除けば、1 日限りのイベントに似ています。参加者がモデルのトレーニングや改良により多くの時間を割けるよう、組織ではこうしたイベントを数日、あるいは何週間も間隔をあけて開催するのが一般的です。ワークショップとレースの合間により多くの時間を設けることで、参加者はレースの当日に激しい競争に集中できます。

- 1 日目: 参加者は、AWS DeepRacer Pit Crew の専門家による対面式の AWS DeepRacer ワークショップ (90 ~ 120 分) に参加します。対面式のワークショップをリクエストする方法の詳細については、「[イベントをリクエストする](#)」を参照してください。
- 2 日目: 参加者はワークショップの後に対面式のレースに参加します (1 回のレースにつき 120 分、または約 5 分)。

カスタムイベントの例

仮想イベントと同様に、カスタムイベントは、複数の場所にちらばるチームのためにイベントを開催する必要のある大規模な組織 (参加者 100 名以上) にとって最適なオプションです。カスタムイベントでは、時間やレース形式に制限がないため、トレーニング、ワークショップ、レースをより柔軟に行うことができます。このようなイベントには、仮想と対人式の両方のレースを含めることができ、これらのレースは、グローバルな参加者がモデルのトレーニングや共同作業に多くの時間を費やすことができるように、複数の週にまたがって実施できます。この種のカスタムイベントは、潜在的な物流面の問題に備えて、事前にいくつかの小規模なイベントを開催した方が成功しやすくなります。この種のイベントや一連のイベントは、組織内で機械学習を宣伝するチームを育成するのにも役立ちます。

- 以下の例では、複数のリージョンにまたがる従業員に対応するため、カスタムイベントを 3 か月にまたがって開催しています。
 - 1 か月目: 複数の場所で開催されるグローバルワークショップ。
 - 2 か月目: 複数のリージョンから参加できる、1 か月にわたる仮想のリーグ予選レース。

- 3 か月目: 対面レースと仮想チャンピオンシップカップのレース。この対面レースは、世界中のチームが観戦できるようにライブ配信することができます。

カスタムイベントのホストに関心がある場合は、AWS に連絡してイベントサポートを受けてください。「[イベントのリクエスト](#)」を参照してください。

その他のリソース

AWS DeepRacer イベントに関連するその他のリソースについては、以下のリストを参照してください。

- [AWS DeepRacer のブログ](#)
- [AWS DeepRacer League](#)
- [AWS DeepRacer コミュニティの Slack チャンネル](#)
- [機械学習トレーニング](#)
- [機械学習の認定](#)
- [AWS DeepRacer トレーニング](#)
- [AWS DeepRacer GitHub リポジトリ](#)
- [レースのヒント](#)
- [AWS DeepRacer の YouTube チャンネル](#)
- [AWS DeepRacer ストアフロント](#)

マルチユーザー モード

マルチユーザーモードのアカウント設定は、組織が 1 つの AWS アカウントで複数の AWS DeepRacer 参加者をスポンサーするエキサイティングな方法を提供します。スポンサー付き参加者は、自分の費用は一切発生しません。代わりに、トレーニング時間とストレージコストがスポンサー AWS アカウントに請求されます。AWS DeepRacer のイベント主催者は、マルチユーザーモードのアカウントを設定することで、参加者個人、グループ、または参加者全体のトレーニング時間やモデルに関するデフォルトのクォータを更新することにより、予算を設定し、支出を監視および管理することができます。

次のセクションでは、管理者または参加者として AWS DeepRacer マルチユーザーモードを利用開始する方法について説明します。

Note

アカウントスポンサー付きのマルチユーザーモードは、AWS DeepRacer サービス内でのみ利用できます。

トピック

- [マルチユーザーモードを設定する \(管理者\)](#)
- [AWS DeepRacer のマルチ ユーザーエクスペリエンス \(参加者\)](#)

マルチユーザーモードを設定する (管理者)

マルチユーザーアカウントを設定すると、主催者 (アカウント管理者など) は、参加者にアカウント ID で AWS DeepRacer へのアクセスを提供できるようになります。また、AWS DeepRacer コンソールから、参加者のトレーニング時間に対する使用量クォータの設定、トレーニングとストレージへの支出の監視、トレーニングの開始と停止、アカウント内のすべてのユーザーのモデルの表示と管理を行うことができます。

マルチユーザーモードは、個々の AWS アカウントを持たない複数の参加者がいる大規模なイベントに特に役立ちます。イベントの各参加者のアカウントを作成して管理する代わりに、AWS DeepRacer 管理者は 1 つの AWS アカウントを通じてすべてのスポンサー参加者をホストできます。

マルチユーザーモードでは、スポンサー参加者は自分のコストを発生させることなく競い、トレーニングすることができます。トレーニングおよびストレージ料金は、スポンサーとなるマルチユーザーの AWS アカウントに請求されます。管理者が、参加者の使用に対する資金提供を停止した場合、参加者はレーサーのエイリアスとプロフィールを保持します。

マルチユーザーの利害関係者

このウォークスルーでは、マルチユーザーモードの設定と使用における一般的なマルチユーザーの利害関係者について説明します。

- AWS IAM/SSO 設定の 管理者。IAM/SSO 設定の AWS 管理者は、AWS DeepRacer 管理者と参加者がマルチユーザーモードを使用するように IAM または SSO を設定します。IAM/SSO の AWS 管理者には、IAM と SSO の管理者権限があります。IAM ユーザーの作成の詳細については、「[AWS アカウントでの IAM ユーザーの作成](#)」を参照してください。
- AWS DeepRacer 管理者 AWS DeepRacer 管理者は AWS DeepRacer 参加者のスポンサーシップを管理し、スポンサーシップの一時停止と再開、モデルおよびアーティファクトの削除、仮想レースの設定とホスト、マルチユーザーモードの有効化および無効化を行うことができます。AWS DeepRacer 管理者には [AWSDeepRacerAccountAdminAccess](#) のアクセス許可があります。
- AWS DeepRacer 参加者 AWS DeepRacer 参加者は、マルチユーザーモードの管理者の AWS アカウントの下でイベントに招待され、イベントに参加します。参加者には [AWSDeepRacerDefaultMultiUserAccess](#) の許可があり、スポンサーのアカウント内でトレーニング、評価、モデルの保存を行えます。参加者はまた、レーサープロフィールを設定し、仮想レースに参入し、物理的な AWS DeepRacer 車両にデプロイするためのモデルをダウンロードします。

このウォークスルーでは、以下のステップを実行します。

- ステップ 1. 前提条件を実行します。
- ステップ 2. AWS DeepRacer アカウントでマルチユーザーモードをアクティブにします。
- ステップ 3. 参加者を招待します。
- ステップ 4. 使用量クォータを設定します。
- ステップ 5. スポンサー付き参加者の使用状況を監視します。

ステップ 1. AWS DeepRacer マルチユーザーモードの前提条件

マルチユーザーモードのための以下の前提条件を満たします。

- [マルチユーザー用の AWS DeepRacer 管理者権限でアカウントを設定します。](#) マルチユーザーモードでレースを組織し、一般的な AWS DeepRacer 管理者タスクを実行する場合は、[AWSDeepRacerAccountAdminAccess](#) 許可を使用して AWS DeepRacer 管理者としてアカウントを設定する必要があります。
- [スポンサーする参加者に AWS コンソールアクセスとレーサーポリシーのアクセス許可を付与します。](#)

マルチユーザー用の AWS DeepRacer 管理者権限でアカウントを設定します。

マルチユーザーモードの AWS DeepRacer 管理者として設定するには、ユーザー、グループ、あるいはロールに添付された IAM AWS DeepRacer 管理者ポリシー [AWSDeepRacerAccountAdminAccess](#) が必要です。組織によっては、コンソールを使用してユーザーまたはロールを作成し、必要な IAM ポリシーを添付して、管理者ポリシーを自分でセットアップするか、IT 管理者に提供してもらうことがあります。必要な管理者ポリシーの詳細については、「[AWSDeepRacerAccountAdminAccess](#)」を参照してください。IAM ポリシーの詳細については、「IAM ユーザーガイド」の「[アクセス管理](#)」を参照してください。

スポンサー付き参加者への AWS コンソールアクセスを提供する

スポンサーとなるレーサーに AWS DeepRacer コンソールへのアクセスを提供するには、[AWS IAM Identity Center](#) や などの標準 AWS 認可プロトコルを使用することをお勧めします [AWS Identity and Access Management](#)。お客様の組織の既存の SSO を通じてアクセスを提供することも可能です。指定した認証情報を使用して AWS DeepRacer コンソールにログインすると、AWS プレイヤーアカウントを作成してログインし、AWS アカウントの AWS DeepRacer コンソールにアクセスするように求められます。AWS プレイヤーアカウントの詳細については、「[AWS プレイヤーアカウント](#)」を参照してください。

IAM を使用してスポンサー付き参加者に AWS コンソールアクセスを提供する

1. 各参加者の IAM ユーザー名とパスワードを作成します。「[AWS アカウントでの IAM ユーザーの作成](#)」にアクセスします。
2. 各参加者に [AWSDeepRacerDefaultMultiUserAccess](#) の権限を付与します。さらなる詳細については、「[AWS の AWS DeepRacer の管理ポリシー](#)」を参照してください。
3. 参加者に IAM ユーザー名とパスワード、コンソールへのリンクを E メールで送信します。提供されたリンクを使用して IAM ユーザー名とパスワードを入力すると、参加者はコンソールにアクセスできます。IAM ユーザーの作成の詳細については、「[AWS アカウントでの IAM ユーザーの作成](#)」を参照してください。

IAM Identity Center を使用してスポンサー付き参加者に AWS コンソールアクセスを提供する

1. <https://console.aws.amazon.com/singlesignon/> の IAM Identity Center コンソールを開き、カスタム権限セットを作成し、ユーザーをアカウントに割り当てます。詳細については、「[のアクセス権限セット](#)」を参照してください。
 2. カスタムアクセス権セットを作成する場合は、次の値を入力します。
- リリーステート: <https://console.aws.amazon.com/deepracer/home?region=us-east-1#getStarted>

Note

リリーステートは、アカウント内の参加者を指定された URL にリダイレクトします。この場合、AWS DeepRacer コンソールにリダイレクトします。

- AWS 管理ポリシー: `AWS DeepRacerDefaultMultiUserAccess`

前提条件を満たした後に、マルチユーザーモードを有効にして、アカウントを通じて参加者をレースするように招待する準備が整います。

ステップ 2: マルチユーザーアカウントモードを有効にします。

AWS DeepRacer 管理者アカウントを設定し、スポンサー参加者にコンソールアクセスとアクセス権限を付与したら、AWS DeepRacer アカウントでマルチユーザーモードをアクティブ化できます。

Note

デフォルトでは、マルチユーザーモードで参加者をスポンサーするアカウントには制限があります。詳細については、「[使用状況のモニタリング](#)」の「アカウントのクォータ」セクションを参照してください。

1. 左のナビゲーションペインで、[マルチユーザー管理] と [設定] ページに移動します。
2. [複数ユーザーアカウントモードを有効にする] で、[マルチユーザーモードを有効にする] をオンにします。
3. [マルチユーザーモードの有効化] ダイアログボックスで、チェックボックスをオンにして、スポンサー付き参加者に必要なアクセスと権限があることを確認します。

4. [マルチユーザーモードの有効化] を選択します。

前提条件を満たし、マルチユーザーモードを有効にすると、スポンサーされた各参加者は、管理者の AWS アカウントに請求されるすべてのトレーニングおよびストレージ料金でレースを作成し、モデルをトレーニングできます。デフォルトでは、1 人の参加者は 3 つの同時参加モデルのクォータを持ち、一度に 10 件までのオープンレースまたは今後のレースを管理することができます (LIVE、クラシック、学生レースを含む)。

マルチユーザーアカウントモードを無効にする

マルチユーザーモードを無効にすると、管理者アカウントの下に新しいプロフィールが作成されず、以前のスポンサー付き参加者のプロフィールが管理者のアカウントに表示されなくなります。参加者は AWS プレイヤーアカウントにログインするよう求められなくなり、管理者のアカウントで作成されたモデルにアクセスしたりトレーニングしたりできなくなります。

管理者は、スポンサー付き参加者のモデルをダウンロード、保存、およびインポートできます。

1. マルチユーザー管理と設定ページに移動します。
2. [マルチユーザーアカウントモードを無効にする] で、マルチユーザーモードを無効にするを選択します。
3. [マルチユーザーモードの無効化] ダイアログボックスで、チェックボックスを選択し、マルチユーザーモードを無効にすることを確認します。[マルチユーザーモードを無効化] を選択する

マルチユーザーモードは無効です。

Note

スポンサー AWS マルチユーザーアカウントで作成されたすべてのモデルは保持され、モデルが削除されるまでモデルストレージコストは AWS アカウントに残ります。

ステップ 3: 参加者をスポンサーに招待する

提供されたメールテンプレートを使用して、参加者をスポンサー付き参加者としてトレーニングおよびレースするように招待できます。

参加者を招待する。

1. 左のナビゲーションペインで、[マルチユーザー管理] と [設定] ページに移動します。[ユーザーの招待] の [マルチユーザーモードの設定] で、[招待テンプレートを見る] を選択します。
2. メールクライアントアプリケーションに表示されるメールテンプレートをコピーし、それを使用して、招待するスポンサー付き参加者に送信するメールを作成します。会社の既存の SSO を使用している場合は、参加者が使用する SSO URL を含めることができます。または、参加者が AWS コンソールにログインするために使用する IAM 認証情報を指定することもできます。

ステップ 4: 使用量クォータを設定する

スポンサー付き参加者が招待メールを受け取り、アカウントでプロフィールを作成すると、[使用状況の監視] 画面の [スポンサー付きユーザー] リストに表示されます。この画面では、スポンサー付き参加者の利用可能なトレーニング時間とモデル数に使用制限を設定できます。クォータを設定することで、アカウントの参加者 1 人あたりのコストを管理し、参加者が使用クォータを超えないようにすることができます。また、必要に応じて使用量クォータを増やしたり減らしたりして、スポンサー付き参加者に、AWS DeepRacer モデルを効果的にトレーニングするのに必要な時間を提供することもできます。

Note

デフォルトでは、マルチユーザーモードでスポンサー付き参加者は 5 時間のトレーニング時間を受け取ります。

スポンサー付きレーサーの使用量クォータを編集するには

1. 左のナビゲーションペインで、[マルチユーザー管理] と [使用状況のモニタリング] ページに移動します。スポンサー付きユーザーの [使用状況のモニタリング] 画面で、クォータを設定する参加者を選択します。[アクション] を選択してドロップダウンリストを開き、[使用量クォータの設定] を選択します。
2. [使用クォータの設定] ポップアップで、選択した参加者の [最大トレーニング時間] と [最大モデル数] を入力します。[確認] を選択して変更を保存するか、[キャンセル] を選択して変更を破棄します。

ステップ 5: 使用状況のモニタリング

推定支出やトレーニングモデル時間など、スポンサー付き参加者の使用状況を監視できます。また、参加者のスポンサーシップを一時停止したり、モデルの削除、使用状況の概要を表示したりできま

す。[使用状況のモニタリング] ページ内の AWS DeepRacer マルチユーザー管理でのモニタリング使用に関するすべてのタスクを実行します。

スポンサー付きレーサーの費用に関するすべての情報は見積もりであり、予算編成や原価計算の目的に使用しないでください。見積もりは米ドルとなり、特別価格には反映されません。料金の詳細については、「[料金](#)」を参照してください。

マルチユーザーモードのアカウントクォータ

デフォルトでは、マルチユーザーモードのスポンサーアカウントには、すべてのスポンサープロフィールで共有される次のクォータがあります。

- 100 件の同時トレーニングジョブの合計数
- 100 件の同時評価ジョブ
- 100 件のオープンレースまたは今後のレース (LIVE、クラシック、学生レースを含む)
- 1000 台の自動車
- 50 台のプライベートリーダーボード

これらのクォータを調整するには、[カスタマーサービス](#)までご連絡ください。

支出の見積もりを表示するには

[使用状況のモニタリング] ページの [使用状況のモニタリング] で、参加者の使用状況の推定概要を表示できます。

請求アラートを設定するには

アカウントの請求アラートを設定できます。請求アラートは、支出を最新の状態に維持するのに役立ちます。詳細については、「[請求に関するよくある質問](#)」を参照してください。

スポンサーシップを一時停止するには

1 人の参加者、複数の参加者、またはすべての参加者のスポンサーシップを一時停止できます。スポンサーシップを一時停止すると、スポンサー参加者はアカウントで新しいモデルを作成したり、モデルをトレーニングしたりすることはできません。進行中のトレーニングは完了まで実行され、支出の見積もりに含まれます。スポンサーシップはいつでも再開することができます。マルチユーザーアクセスが一時停止された参加者は、モデルを表示したり、モデルをリーダーボードに投稿したりすることはできませんが、コストを発生させるアクティビティは実行できません。

1. 「使用状況のモニタリング」ページの [使用状況のモニタリング] の下にある [スポンサー付きユーザー] セクションで、スポンサーシップを一時停止するユーザーを選択します。
2. [スポンサーシップの一時停止] を選択します。
3. [スポンサーシップの一時停止] ダイアログボックスで、[スポンサーシップの一時停止] を選択してスポンサーシップを一時停止します。スポンサーシップを一時停止したくないと判断した場合、[キャンセル] を選択します。

スポンサーシップを再開するには

スポンサーシップを一時停止したレーサーに対するスポンサーシップを再開できます。

1. [使用状況のモニタリング] ページの [使用状況のモニタリング] の下にある [スポンサー付きユーザー] セクションで、スポンサーシップを一時停止するユーザーを選択します。
2. [スポンサーシップを再開する] を選択します。

レーサーのモデルを表示する方法

- 「モデルの作成」ページの [モデル] から、モデルとユーザーのモデルを表示できます。

次の手順

マルチユーザーモードを設定し、アクティブ化したら、以下の手順を実行できます。

- コミュニティレースを作成します。
- AWS DeepRacer ワークショップをリクエストします。

コミュニティレースを作成

コミュニティレースは、スポンサー付き参加者が強化学習を体験するためのエキサイティングな方法を提供します。

コミュニティレースを作成し、スポンサー付き参加者を招待できます。

詳細については、「[the section called “レースを作成する \(クイックスタート\)”](#)」を参照してください。

ワークショップをリクエストする

60 分間のオンラインまたは対面ワークショップで AWS DeepRacer の詳細を学ぶためのワークショップをリクエストできます。

詳細については、「[ワークショップ](#)」を参照してください。

AWS DeepRacer のマルチ ユーザーエクスペリエンス (参加者)

このチュートリアルは、個人のプロフィールが、マルチユーザー モードの組織のアカウントによってスポンサーされている参加者の体験を示すものです。

AWS DeepRacer は、AWS DeepRacer モデルのトレーニングやレースをすることによって、強化学習 (RL) を体験できるエキサイティングな方法を提供します。組織では、自分の AWS アカウントでプロフィールをスポンサーする機会を提供する場合があります。モデルのトレーニング、評価、保存など、生成したすべての料金は、ログインに使用した AWS アカウントに請求されます。プロフィールをスポンサーする AWS アカウントの管理者は、モデル、車、リーダーボードを表示したり、トレーニング時間を一時停止したり、トレーニング時間とストレージクォータを調整したり、プロフィールのスポンサーを停止したりできます。

スポンサー付きレーサーのサインアップ プロセスの一環として、AWS プレイヤーアカウントを作成します。アカウントは、保持するポータブルプロフィールであり、他の多くの AWS サービスで使用できます。詳細については、「[AWS プレイヤー アカウント](#)」を参照してください。

前提条件

組織のイベントコーディネーターは、AWS コンソールのログイン認証情報を含む AWS DeepRacer への参加の招待を共有します。コンソールにログインするには、これらの認証情報を使用します。また、セットアップの一環としてレーサープロフィールと AWS プレイヤーアカウントを作成します。

このチュートリアルは、以下のステップを含みます。

- スポンサーアカウントの認証情報を使用して AWS コンソールにログインします。
- AWS Player アカウントを作成またはログインします。
- プロフィールをカスタマイズします。
- モデルをトレーニングします。
- スポンサーの利用状況を表示します。

- (オプション) スポンサー時間の追加をリクエストします。

ステップ 1. スポンサーアカウントの認証情報を使用して AWS コンソールにログインする

スポンサー参加者として、AWS DeepRacer をスタートするには、イベントコーディネーターから受け取った招待状に記載されている認証情報を使用して、コンソールにログインします。

スポンサー参加者として AWS コンソールにログインするには

1. イベントコーディネーターから受け取った招待状に記載されている認証情報を使用します。
2. コンソールで、 に移動します AWS DeepRacer。

AWS Player アカウントページが表示されます。

ステップ 2. AWS Player アカウントを作成またはログインする

1. AWS プレイヤーアカウント ページで、プレイヤーアカウントの新規作成または既存の AWS プレイヤーアカウントにログインします。
 - アカウントをまだ持っていない場合、[アカウントの作成] を選択、E メールアドレスとパスワードを入力し、[アカウントを作成する] を選択します。
 - Player AWS アカウントを既にお持ちの場合は、E メールとパスワードを入力し、サインインを選択します。
2. アカウントの設定を有効にするためのメッセージが、指定したメールアドレスに送信されます。
3. E メールで受け取った検証コードを、[検証コード] ボックスに入力し、[登録の確認] を選択します。

Note

検証コードを入力するまで、現在のページを表示したままにします。

これでスポンサー参加者として、AWS DeepRacer コンソールにログインしました。

4. ステップ 3 に進み、レーサープロフィールをカスタマイズします。

ステップ 3. プロフィールのカスタマイズ。

プロフィールイメージの編集や、レーサーネームを追加して、プロフィールをカスタマイズします。レーサープロフィールはいつでも更新および変更ができます。また、AWS DeepRacer League で獲得した賞品に関する連絡を受け取るため、居住国と連絡先のメールアドレスを追加することもできます。さらに、AWS DeepRacer League で受け取ったパフォーマンスの実績は、[レースのプロフィール] ページからソーシャルメディアで共有できます。

Note

AWS DeepRacer League のレースイベントへの参加やモデルをトレーニングするには、レーサーネームを作成し、居住国を追加する必要があります。レーサーネームは世界で唯一のものでなければなりません。居住国を選択すると、レースシーズンの期間中は固定されます。

プロフィールイメージをカスタマイズします。

1. 左のナビゲーションペインで、「レーサープロフィール」のページに移動します。
2. 「レーサープロフィール」ページで、[編集] を選択します。
3. 「レーサープロフィール」のダイアログボックスで、ドロップダウンリストから項目を選択して、プロフィールイメージをカスタマイズします。
4. [保存] を選択します。

レーサーネームをカスタマイズします。

1. 左のナビゲーションペインで、「レーサープロフィール」のページに移動します。
2. 「レーサープロフィール」のページで、[編集] を選択します。
3. [レーサーのプロフィール] のダイアログボックスで、[レーサーネームを変更する] を選択し、プロフィールネームを入力します。
4. [保存] を選択します。

ステップ 4. モデルをトレーニングします。

プロフィールのカスタマイズが完了したら、モデルのトレーニングをスタートします。詳細については、「[AWS DeepRacer モデルのトレーニングと評価](#)」を参照してください。

ステップ 5. スポンサーの利用状況を表示します

スポンサー時間やモデルを記録しておき、それらを最大限に活用できるようにしましょう。

スポンサー時間の利用状況と保存されたモデルを表示するには

- 「レーサーのプロフィール」ページの [スポンサー時間の利用状況] で、利用した合計時間をご覧ください。

ステップ 6. (オプション) スポンサー時間の追加をリクエストします

スポンサー付きの参加者には、5 時間の無料トレーニング時間が付与されます。無料のスポンサー時間を使い切ると、アカウント管理者またはイベント主催者に追加時間をリクエストできます。代わりに、追加のスポンサー時間を利用できない場合は、独自の AWS DeepRacer アカウントを作成して AWS DeepRacer を引き続きご利用いただくこともできます。トレーニングやストレージ費用の詳細については、「[料金表](#)」を参照してください。

AWS DeepRacer Student の教育者向けツール

このセクションでは、AWS DeepRacer 機械学習カリキュラムを教室に組み込んだり、AWS DeepRacer Student の実践ラボを開催したり、学生コミュニティレースを開催したりするための情報とリソースを提供します。

AWS DeepRacer Student を教室に組み込む

AWS DeepRacer を使い始めたばかりの教育者の方は、AWS DeepRacer Student 教育者向けプレイブックを読むことをお勧めします。

カリキュラムプレイブック

「[AWS DeepRacer Student カリキュラムプレイブック](#)」には、各 AWS DeepRacer Student モジュールの概要、学習目標、学習成果、主要な概念、サポート資料、評価とアクティビティの提案に関する概要が記載されています。

学生ラボプレイブック

「[AWS DeepRacer Student ラボプレイブック](#)」には、教育者が AWS DeepRacer Student の実践ラボを開催するための情報とリソースが記載されています。実践ラボは、AWS DeepRacer Student League レース、プライベートコミュニティレース、LIVE 仮想レーシングなどの仮想イベントや、物理的なトラックと AWS DeepRacer デバイスを使用した対面式イベントで構成されます。

学生コミュニティレースの作成

教育者向けプレイブックを使い始めたら、「[AWS DeepRacer コンソール](#)」の「コミュニティレース」を使用して、「[AWS DeepRacer Student League](#)」の学生向けのレースを作成します。レース招待のリンクを共有して、学生レース参加者を招待します。

教育者は、レースを作成および整理するために AWS DeepRacer コンソールにサインインするための AWS アカウントが必要ですが、学生は AWS DeepRacer Student League へのログイン、プロフィールの更新、無料コースの開始、AWS DeepRacer モデルの作成にのみ E メールアドレスが必要です。教育者は、メールアドレスを使用して AWS DeepRacer Student League にアカウントを作成し、カリキュラムをプレビューしたり、レース体験を試したり、学生の進捗状況を監視したりすることもできます。

AWS DeepRacer Student 仮想レースを作成または管理するには、以下のトピックのいずれかを参照してください。

トピック

- [the section called “学生レースの作成”](#)
- [the section called “学生レースのカスタマイズ”](#)
- [the section called “学生レースの管理”](#)

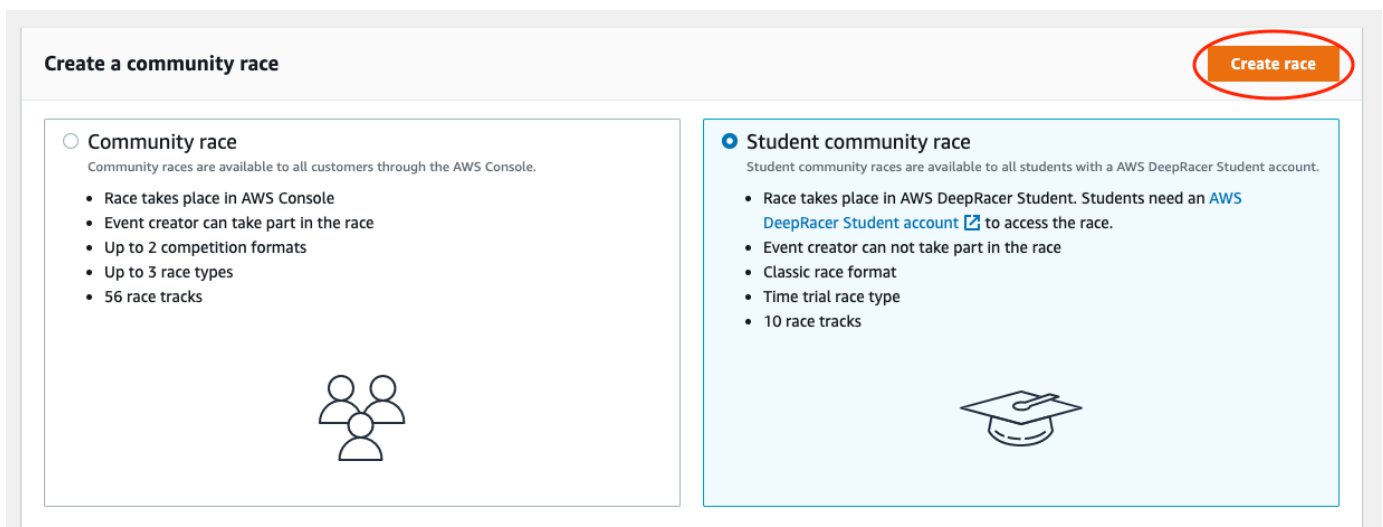
AWS DeepRacer Student コミュニティレースを作成する

デフォルトの学生コミュニティレースの設定を使用して、仮想レースをすばやくセットアップできます。

学生コミュニティレースは、リアルタイムの対応を必要としない非同期型イベントです。レースにモデルを提出し、リーダーボードを表示するには、参加者は招待リンクを受け取る必要があります。レーサーは、日付範囲内でいつでも無制限のモデルを提出して、リーダーボードに登ることができません。送信されたモデルのクラシックレースの結果と映像は、リーダーボードのページでレース開始直後から確認できます。

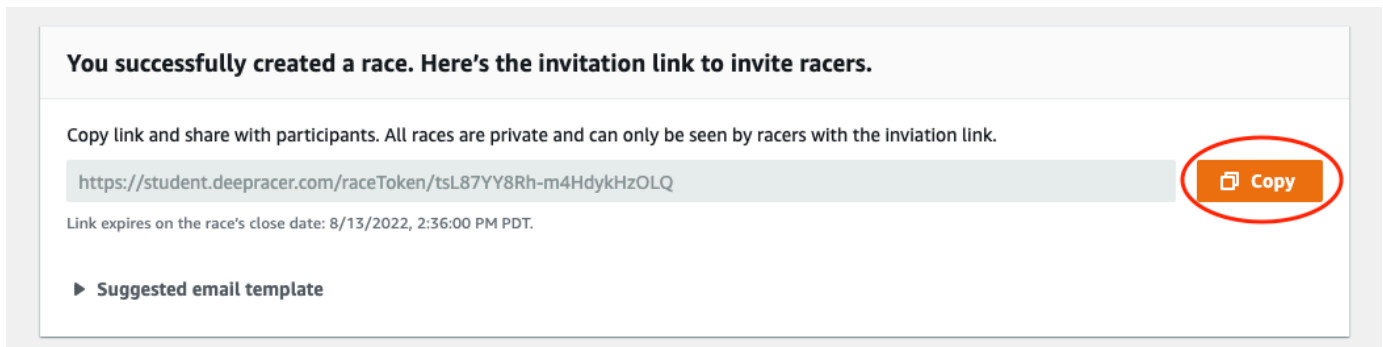
学生コミュニティレースの作成を開始するには

1. [\[AWS DeepRacer コンソール\]](#) を開きます。
2. [コミュニティレース] ページで [学生コミュニティレース] を選択します。
3. [スタックの作成] をクリックします。



4. レースには、オリジナルでわかりやすい名前を入力します。
5. イベントの開始日時を 24 時間形式で指定します。AWS DeepRacer コンソールは、タイムゾーンを自動的に認識します。終了日時も入力します。

6. デフォルトのレース設定を使用するには、[次へ] を選択してください。すべてのオプションについて学ぶ準備ができたなら、「[the section called “学生レースのカスタマイズ”](#)」を参照してください。
7. レースの詳細を確認するには、「レースの詳細」ページで、レースの仕様を確認してください。設定を変更するには、「レースの詳細」ページで、[編集] または [戻る] に戻ります。招待リンクを取得する準備ができたなら、[送信] を選択します。
8. レースを共有するには、[コピー] を選択し、メールテンプレート、テキストメッセージ、お気に入りのソーシャルメディア アプリケーションにペーストします。クラシックレースは、招待リンクのあるレーサーのみ見ることができます。リンクはレースの終了日に期限切れになります。



9. 学生レースのタイムフレームが近づいたら、[レースを管理する] ページの [レーサー] で、誰がモデルに参入したのか、誰がまだそうする必要があるので書き留めておきましょう。

[[レースを管理する](#)] を選択し、選択したレーストラックを変更し、レースの説明を追加し、ランキング方法を選択し、レーサーが許可されるリセットの数を決定し、RL モデルが自分のレースの資格を得るために完了しなければならない最小ラップ数を決定し、トラック外ペナルティを設定し、その他のレースの詳細をカスタマイズしてください。

Note

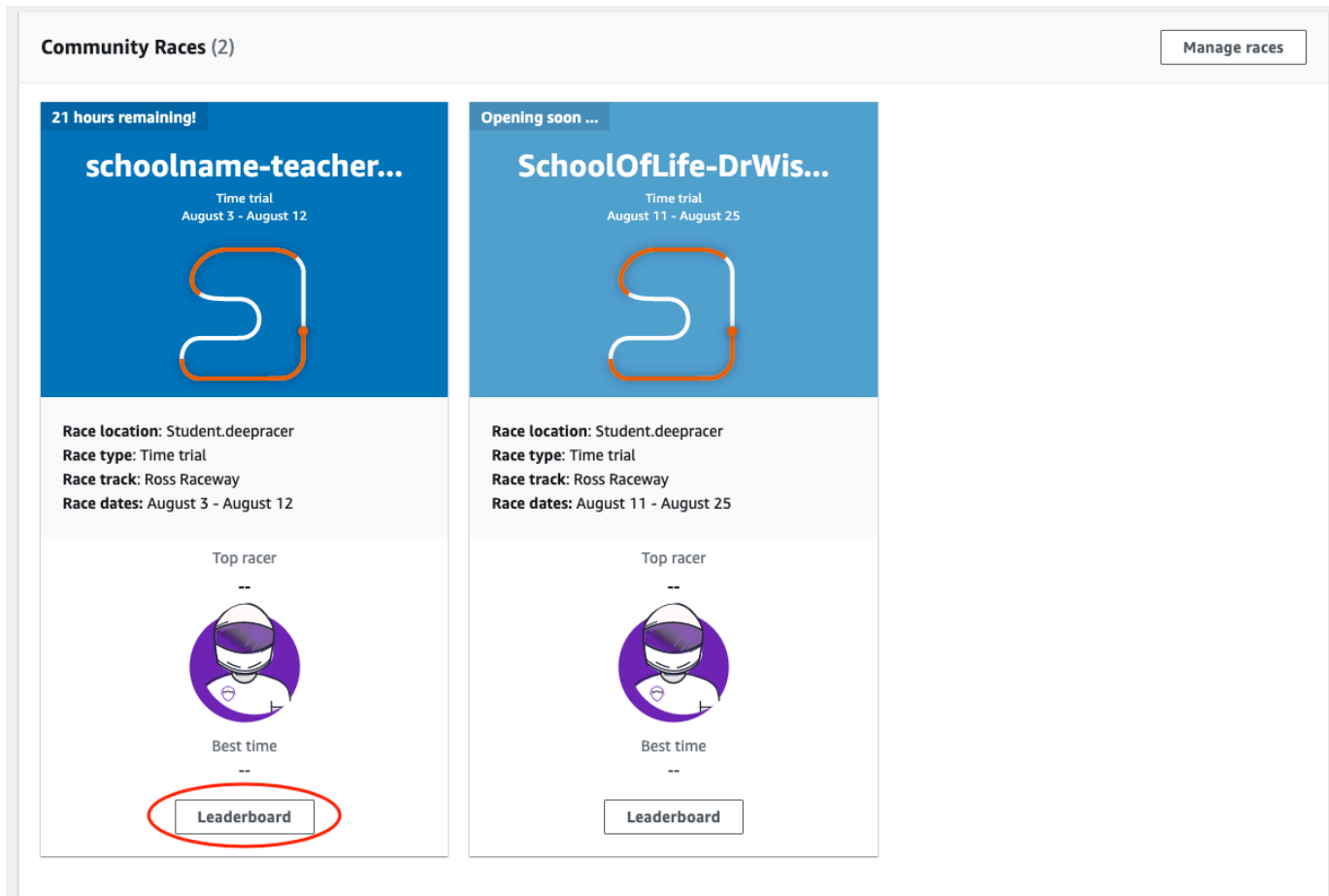
学生のエイリアスは、[レーサー] タブと [リーダーボード] にのみ表示されるので、どのエイリアスがどの学生に関連付けられているかを書き留めておいてください。

AWS DeepRacer Student コミュニティレースをカスタマイズする

グループに合わせたレースを作成するには、レースの複雑さや難易度を高めるカスタマイズを追加します。

学生レースのカスタマイズ

1. [\[AWS DeepRacer コンソール\]](#) を開きます。
2. [\[コミュニティレース\]](#) を選択します。
3. [\[コミュニティレース\]](#) ページで、カスタマイズしたいレースの [\[リーダーボード\]](#) を選択します。




4. [\[レースの詳細\]](#) ページで、[\[レースの編集\]](#) を選択します。

AWS DeepRacer > Community races > SchoolOfLife-DrWise

SchoolOfLife-DrWise Info

[Edit race](#)

Race details

<p>Race location Student.deepracer</p> <p>Race hosting Classic race</p> <p>Race type Time trial</p> <p>Race dates Start August 11, 2022 at 11:27 PM End August 25, 2022 at 12:00 PM</p> <p>Time zone UTC-0700 (Pacific Daylight Time) America/Los_Angeles</p>	<p>Competition track The Ross Raceway was named in honor of the 2021 AWS DeepRacer 3rd place Champion, Ross Williams. Expect to see racers put the pedal to the metal on the 3x dragstrips featured on the Ross Raceway, but there will be no shortage of off tracks as they navigate the various sweeping turns.</p> 	<p>Rules</p> <p>Ranking method Total time</p> <p>Style Individual lap</p> <p>Entry criteria 3 consecutive laps</p> <p>Resets Unlimited resets</p> <p>Off-track penalty 3 seconds</p>
--	--	--

5. [レースのカスタマイズ] を展開します。

Search for services, features, blogs, docs, and more [Option+S] N. Virginia

Race customizations

Description of the racing event- optional
The race description will only be displayed under the race details in the AWS DeepRacer console.

Summarize the goals and rules of the event for participants.

Ranking method
Choose between Best lap time (the winner is the racer who posts the fastest lap) or Total time (the winner is the racer with the fastest overall average).

Total time

Minimum laps
Choose the number of laps required for a model to pass evaluation.

3 consecutive laps

Off-track penalty
Choose the number of seconds added to a racer's time for driving off track.

3 seconds

Community races visibility
Races are private. Only racers that are invited to a race can view it. To invite racers to your race, you share a link. Racers you've invited can forward the link to other racers. As the race organizer, you can revoke any racer's permission to race.

6. 必要に応じて、参加者のイベントの目標とルールを要約したレースの説明を作成します。説明はリーダーボードの詳細に表示されます。

7. クラシックレースの [ランキング方法] では、勝者が最速ラップを記録するレーサーならば [ベストラップタイム] から選びます。[平均時間] を選ぶと、イベントの時間枠内に複数回の試行の後、勝者は平均時間が最も短いレーサーになります。あるいは [合計時間] を選ぶと、勝者は総合平均が最も短いレーサーになります。
8. クラシックレースの場合は、[最小ラップ] の値を選択します。これはレースのリーダーボードに結果を提出するためにレーサーが完了しなければならない連続ラップ数になります。初心者のレースでは、小さめの数を選びます。上級ユーザーの場合は、大きめの数を使用します。
9. [場外ペナルティ] を使用する場合、RL モデルがトラックから外れたときに、レーサーの時間に加算する秒数を選択します。
10. これで、学生コミュニティレースのすべてのカスタマイズオプションが完了しました。[次へ] を選択して、レースの詳細を確認します。
11. 「レースの詳細を確認する」ページで、レースの仕様を確認してください。設定を変更するには、「レースの詳細」ページで [編集] または [戻る] を選択します。招待リンクを取得する準備ができたなら、[送信] を選びます。
12. [完了] を選択します。「レースを管理する」ページが表示されます。

メールテンプレートを使用して新しいレーサーを招待する方法、レースからレーサーを削除する方法、レーサーのモデル提出状況を確認する方法については、「[コミュニティレースの管理](#)」を参照してください。

AWS DeepRacer Student コミュニティレースに参加する

学生コミュニティレースはすべて、招待リンクを受け取った個人にのみ表示されます。参加者は招待リンクを自由に転送できます。レースに参加するには、[AWS DeepRacer Student](#) アカウントが必要です。初めてのユーザーは、レースに参加する前にアカウントの作成プロセスを完了させる必要があります。学生がアカウントを設定するのに必要なのはメールアドレスのみです。

レース主催者は以下の操作を実行できます。

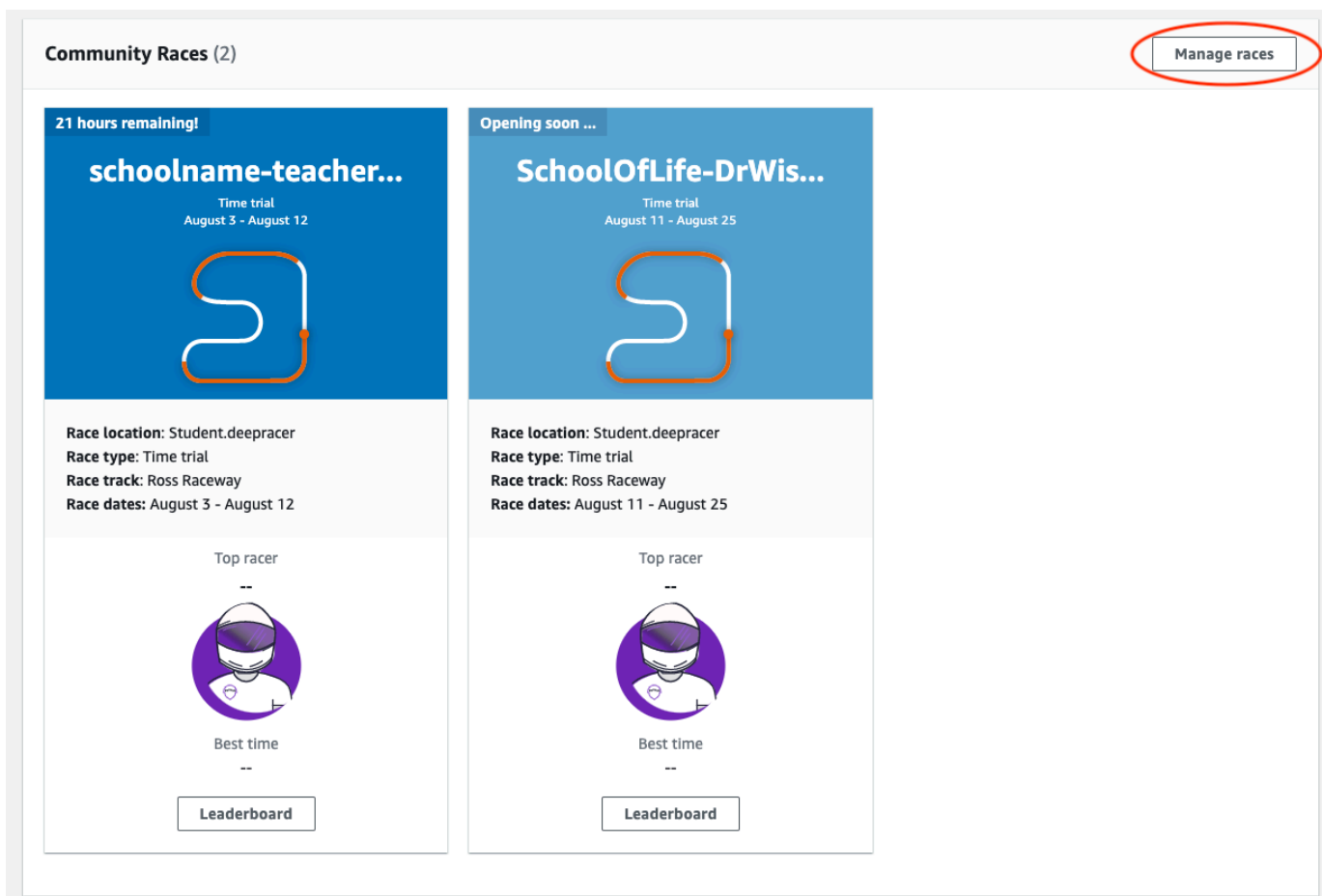
- レースの詳細 (開始日と終了日を含む) を編集
- 参加者を削除
- レースを終了
- レースを削除

Note

学生のエイリアスは、[レーサー] タブと [リーダーボード] にのみ表示されるので、どのエイリアスがどの学生に関連付けられているかを書き留めておいてください。

AWS DeepRacer Student コミュニティレースに参加するには

1. AWS DeepRacer コンソールにサインインします。
2. [コミュニティレース] を選択する。
3. [同期を管理] を選択します。



4. 「レースを管理する」ページで、レースのために自分が管理するレースを選択します。
5. [レースの詳細] を選択し、[編集] を選択します。

AWS DeepRacer > Community races > Manage races

Manage races Info

Races (21) Refresh **Race details** Create race

Search race name

Name	Status	Competition format	Start date	End date
<input checked="" type="radio"/> schoolname-teachername-2022	Opening soon	Classic	8/3/2022, 6:40:00 PM PDT	8/12/2022, 12:00:00 P
<input type="radio"/> AnotherRace	Closed	Classic	9/25/2020, 4:40:00 AM PDT	9/25/2020, 12:00:00 P
<input type="radio"/> EasyRace	Closed	Classic	9/26/2020, 12:00:00 PM PDT	9/26/2020, 4:00:00 PM

- イベントのリーダーボードを表示するには、[レースの表示] を選択します。
- イベントの招待リンクをリセットするには、[招待リンクのリセット] を選択します。招待リンクをリセットすると、当社のリンクを選択してもレースに参加できなくなります。招待リンクをリセットしても、レースの既存の参加者に影響はありません。
- レースを終了するには、[レースを終了] を選択します。このアクションは直ちにレースを終了します。
- イベントを削除するには、[レースを削除] を選択します。これにより、AWS コンソールと AWS DeepRacer Student からこのレースが完全に削除されます。
- 参加者を削除するには、[レーサー] タブを選択し、1人以上の参加者を選択して [レーサーを削除] を選択します。参加者をイベントから削除すると、その参加者はレースに参加できなくなります。

aws Services Search for services, features, blogs, docs, and more [Option+S] N. Virginia

AWS DeepRacer > Student community races > Manage races > schoolname-teachername-2022

schoolname-teachername-2022 Info

Race details **Racers** View race

Racers (1) Export to CSV Remove racer

Search racers

Alias	Date joined
<input type="checkbox"/> DebbyC123	8/2/2022, 2:32:00 PM PDT

AWS DeepRacer のセキュリティ

AWS DeepRacer を使用して強化学習をトレーニングおよび評価するには、AWS アカウントに、トレーニングジョブを実行する Amazon VPC や、トレーニングされたモデルアーティファクトを保存する Amazon S3 バケットなど、依存する AWS リソースにアクセスするための適切なセキュリティアクセス許可が必要です。

AWS DeepRacer コンソールには、依存サービスに必要なセキュリティ設定をセットアップするための方法が用意されています。このセクションでは、AWS DeepRacer が依存する AWS サービス、および依存サービスにアクセスするために必要なアクセス許可を定義する IAM ロールとポリシーについて説明します。

トピック

- [AWS DeepRacer でのデータ保護](#)
- [AWS DeepRacer 依存 AWS サービス](#)
- [AWS DeepRacer が依存 AWS サービスを呼び出すために必要な IAM ロール](#)
- [AWS Identity and Access Management for AWS DeepRacer](#)

AWS DeepRacer でのデータ保護

AWS DeepRacer は、AWS [責任共有モデル](#)。には、データ保護に関する規制とガイドラインが含まれています。AWS は、すべての AWS サービスを実行するグローバルインフラストラクチャを保護する責任があります。AWS は、このインフラストラクチャでホストされるデータの制御を維持します。顧客コンテンツと個人データを処理するためのセキュリティ設定コントロールを含めます。AWS 顧客および APN パートナー。データコントローラーまたはデータ処理者として動作するは、AWS クラウドに保存した個人データについて責任を負います。

データ保護の目的で、AWS アカウントの認証情報を保護し、AWS Identity and Access Management (IAM) を使用して個々のユーザーアカウントを設定することをお勧めします。これにより、各ユーザーに各自の職務を果たすために必要なアクセス許可のみが付与されます。また、次の方法でデータを保護することもお勧めします:

- 各アカウントで多要素認証 (MFA) を使用します。
- SSL/TLS を使用して AWS リソースと通信します。
- で API とユーザーアクティビティのログ記録を設定します AWS CloudTrail。

- AWS 暗号化ソリューションと、サービス内のすべての AWS デフォルトのセキュリティコントロールを使用します。
- Amazon Macie などの高度なマネージドセキュリティサービスを使用します。これにより、Amazon S3 に保存される個人データの検出と保護が支援されます。

顧客のアカウント番号などの機密の識別情報は、[名前] フィールドなどの自由形式のフィールドに配置しないことを強くお勧めします。これは、コンソール、API、または SDK を使用して AWS DeepRacer AWS CLI または他の AWS のサービスを使用する場合も同様です。AWS SDKs お客様が AWS DeepRacer あるいは他のサービスに入力したあらゆるデータは、診断ログの内容として取得される可能性があります。外部サーバーへの URL を指定するときは、そのサーバーへのリクエストを検証するための認証情報を URL に含めないでください。

データ保護の詳細については、AWS セキュリティブログ のブログ投稿「[AWS の責任共有モデルと GDPR](#)」を参照してください。

AWS DeepRacer 依存 AWS サービス

AWS DeepRacer は、以下の AWS のサービス を使用して必要なリソースを管理します。

Amazon Simple Storage Service

トレーニング済みのモデルのアーティファクトを Amazon S3 バケットに保存する方法。

AWS Lambda

報酬関数を作成して実行する方法

AWS CloudFormation

AWS DeepRacer モデルのトレーニングジョブを作成する方法。

SageMaker AI

AWS DeepRacer モデルをトレーニングする方法

依存、AWS CloudFormation、および SageMaker AI は AWS Lambda、Amazon CloudWatch や Amazon CloudWatch Logs などの他の AWS サービスを使用します。

次の表は、AWS DeepRacer が直接または間接的に使用する AWS サービスを示しています。

AWS AWS DeepRacer が直接または間接的に使用するサービス

AWS サービスプリンシパル	コメント
<u>application-autoscaling</u>	<ul style="list-style-type: none"> オペレーションを自動的にスケーリングするために SageMaker AI によって間接的に呼び出されます。
<u>cloudformation</u>	<ul style="list-style-type: none"> AWS DeepRacer から直接呼び出されて、アカウントリソースを作成します。
<u>cloudwatch</u>	<ul style="list-style-type: none"> オペレーションをログに記録するために直接 AWS DeepRacer に呼び出されます。 オペレーションをログに記録するために SageMaker AI によって間接的に呼び出されます。
<u>ec2</u>	<ul style="list-style-type: none"> トレーニングジョブを作成および実行するために、AWS CloudFormation と SageMaker AI によって間接的に呼び出されます。
<u>kinesisvideo</u>	<ul style="list-style-type: none"> キャッシュされたトレーニングストリームを表示するために直接 AWS DeepRacer に呼び出されます。
<u>lambda</u>	<ul style="list-style-type: none"> 報酬関数を作成および実行するために直接 AWS DeepRacer に呼び出されます。
<u>logs</u>	<ul style="list-style-type: none"> オペレーションをログに記録するために直接 AWS DeepRacer に呼び出されます。 オペレーションをログに記録する AWS Lambda ために によって間接的に呼び出されます。
<u>s3</u>	<ul style="list-style-type: none"> SageMaker AI 固有のストレージオペレーションを実行するために SageMaker AI によって間接的に呼び出されます。 "deepracer " で始まる名前のバケットを作成、一覧表示、および削除するために直接

AWS サービスプリンシパル	コメント
sagemaker	<p>AWS DeepRacer に呼び出されます。バケットからオブジェクトをダウンロード、バケットにオブジェクトをアップロード、またはバケットからオブジェクトを削除するためにも呼び出されます。</p> <ul style="list-style-type: none"> 強化学習モデルをトレーニングするために AWS DeepRacer から直接呼び出されます。

AWS DeepRacer を使用してこれらのサービスを呼び出すには、必要なポリシーを添付した適切な IAM ロールが必要です。これらのポリシーとロールの詳細については、「[AWS DeepRacer が依存 AWS サービスを呼び出すために必要な IAM ロール](#)」を参照してください。

AWS DeepRacer が依存 AWS サービスを呼び出すために必要な IAM ロール

モデルを作成する前に、AWS DeepRacer コンソールを使用してアカウントのリソースを設定します。これにより、AWS DeepRacer コンソールで以下の IAM ロールが作成されます。

[AWSDeepRacerServiceRole](#)

AWS DeepRacer がユーザーに代わって必要なリソースを作成し、AWS サービスを呼び出すことを許可します。

[AWSDeepRacerSageMakerAccessRole](#)

Amazon SageMaker AI がユーザーに代わって必要なリソースを作成し、AWS サービスを呼び出すことを許可します。

[AWSDeepRacerLambdaAccessRole](#)

AWS Lambda 関数がユーザーに代わって AWS サービスを呼び出すことを許可します。

[AWSDeepRacerCloudFormationAccessRole](#)

AWS CloudFormation がユーザーに代わって AWS スタックとリソースを作成および管理できるようにします。

リンクに従って、IAM AWS コンソールで詳細なアクセス許可を表示します。

AWS Identity and Access Management for AWS DeepRacer

AWS Identity and Access Management (IAM) は、管理者が AWS リソースへのアクセスを安全に制御 AWS のサービス するのに役立つです。IAM 管理者は、誰を認証 (サインイン) し、誰に DeepRacer リソースの使用を許可する (アクセス許可を持たせる) かをコントロールします。IAM は、追加料金なしで使用できる AWS のサービス です。

トピック

- [オーディエンス](#)
- [アイデンティティを使用した認証](#)
- [ポリシーを使用したアクセスの管理](#)
- [How AWS DeepRacer と IAM の連携](#)
- [AWS DeepRacer のアイデンティティベースのポリシーの例](#)
- [AWS AWS DeepRacer の マネージドポリシー](#)
- [サービス間での不分別な代理処理の防止](#)
- [トラブルシューティング AWS DeepRacer アイデンティティとアクセス](#)

オーディエンス

AWS Identity and Access Management (IAM) の使用方法は、ロールによって異なります。

- サービスユーザー - 機能にアクセスできない場合は、管理者にアクセス許可をリクエストします (「[トラブルシューティング AWS DeepRacer アイデンティティとアクセス](#)」を参照)。
- サービス管理者 - ユーザーアクセスを決定し、アクセス許可リクエストを送信します (「[How AWS DeepRacer と IAM の連携](#)」を参照)
- IAM 管理者 - アクセスを管理するためのポリシーを作成します (「[AWS DeepRacer のアイデンティティベースのポリシーの例](#)」を参照)

アイデンティティを使用した認証

認証とは、ID 認証情報 AWS を使用して にサインインする方法です。、IAM ユーザー AWS アカウ
ントのルートユーザー、または IAM ロールを引き受けることで認証される必要があります。

AWS IAM Identity Center (IAM Identity Center)、シングルサインオン認証、Google/Facebook 認証情報などの ID ソースからの認証情報を使用して、フェデレーテッド ID としてサインインできます。サインインの詳細については、「AWS サインイン ユーザーガイド」の「[AWS アカウントにサインインする方法](#)」を参照してください。

プログラムによるアクセスの場合、は SDK と CLI AWS を提供してリクエストを暗号化して署名します。詳細については、「IAM ユーザーガイド」の「[API リクエストに対するAWS 署名バージョン 4](#)」を参照してください。

AWS アカウント ルートユーザー

を作成するときは AWS アカウント、すべての AWS のサービス および リソースへの完全なアクセス権を持つ AWS アカウント ルートユーザーと呼ばれる 1 つのサインインアイデンティティから始めます。日常的なタスクには、ルートユーザーを使用しないことを強くお勧めします。ルートユーザー認証情報を必要とするタスクについては、「IAM ユーザーガイド」の「[ルートユーザー認証情報が必要なタスク](#)」を参照してください。

IAM ユーザーとグループ

[IAM ユーザー](#)は、特定の個人やアプリケーションに対する特定のアクセス許可を持つアイデンティティです。長期認証情報を持つ IAM ユーザーの代わりに一時的な認証情報を使用することをお勧めします。詳細については、IAM ユーザーガイドの「[ID プロバイダーとのフェデレーションを使用してアクセスする必要がある AWS](#)」を参照してください。

[IAM グループ](#)は、IAM ユーザーの集合を指定し、大量のユーザーに対するアクセス許可の管理を容易にします。詳細については、「IAM ユーザーガイド」の「[IAM ユーザーに関するユースケース](#)」を参照してください。

IAM ロール

[IAM ロール](#)は、特定のアクセス許可を持つアイデンティティであり、一時的な認証情報を提供します。[ユーザーから IAM ロール \(コンソール\) に切り替えるか、または API オペレーションを呼び出すことで、ロールを引き受けることができます。](#) AWS CLI AWS 詳細については、「IAM ユーザーガイド」の「[ロールを引き受けるための各種方法](#)」を参照してください。

IAM ロールは、フェデレーションユーザーアクセス、一時的な IAM ユーザーのアクセス許可、クロスアカウントアクセス、クロスサービスアクセス、および Amazon EC2 で実行するアプリケーションに役立ちます。詳細については、IAM ユーザーガイドの [IAM でのクロスアカウントリソースアクセス](#) を参照してください。

ポリシーを使用したアクセスの管理

でアクセスを制御する AWS には、ポリシーを作成し、ID AWS またはリソースにアタッチします。ポリシーは、アイデンティティまたはリソースに関連付けられたときにアクセス許可を定義します。は、プリンシパルがリクエストを行うときにこれらのポリシー AWS を評価します。ほとんどのポリシーは JSON ドキュメント AWS としてに保存されます。JSON ポリシードキュメントの詳細については、「IAM ユーザーガイド」の「[JSON ポリシー概要](#)」を参照してください。

管理者は、ポリシーを使用して、どのプリンシパルがどのリソースに対して、どのような条件でアクションを実行できるかを定義することで、誰が何にアクセスできるかを指定します。

デフォルトでは、ユーザーやロールにアクセス許可はありません。IAM 管理者は IAM ポリシーを作成してロールに追加し、このロールをユーザーが引き受けられるようにします。IAM ポリシーは、オペレーションの実行方法を問わず、アクセス許可を定義します。

アイデンティティベースのポリシー

アイデンティティベースのポリシーは、アイデンティティ (ユーザー、グループ、またはロール) にアタッチできる JSON アクセス許可ポリシードキュメントです。これらのポリシーは、アイデンティティがどのリソースに対してどのような条件下でどのようなアクションを実行できるかを制御します。アイデンティティベースポリシーの作成方法については、IAM ユーザーガイドの [カスタマー管理ポリシーでカスタム IAM アクセス許可を定義する](#) を参照してください。

アイデンティティベースのポリシーは、インラインポリシー (単一の ID に直接埋め込む) または管理ポリシー (複数の ID にアタッチされたスタンドアロンポリシー) にすることができます。管理ポリシーとインラインポリシーのいずれかを選択する方法については、「IAM ユーザーガイド」の「[管理ポリシーとインラインポリシーのいずれかを選択する](#)」を参照してください。

リソースベースのポリシー

リソースベースのポリシーは、リソースに添付する JSON ポリシードキュメントです。例としては、IAM ロール信頼ポリシーや Amazon S3 バケットポリシーなどがあります。リソースベースのポリシーをサポートするサービスでは、サービス管理者はポリシーを使用して特定のリソースへのアクセスを制御できます。リソースベースのポリシーでは、[プリンシパルを指定する](#) 必要があります。

リソースベースのポリシーは、そのサービス内にあるインラインポリシーです。リソースベースのポリシーでは、IAM の AWS マネージドポリシーを使用できません。

アクセスコントロールリスト (ACL)

アクセスコントロールリスト (ACL) は、どのプリンシパル (アカウントメンバー、ユーザー、またはロール) がリソースにアクセスするためのアクセス許可を持つかを制御します。ACL はリソーススペースのポリシーに似ていますが、JSON ポリシードキュメント形式は使用しません。

Amazon S3、および Amazon VPC は AWS WAF、ACLs。ACL の詳細については、Amazon Simple Storage Service デベロッパーガイドの [アクセスコントロールリスト \(ACL\) の概要](#) を参照してください。

その他のポリシータイプ

AWS は、より一般的なポリシータイプによって付与されるアクセス許可の最大数を設定できる追加のポリシータイプをサポートしています。

- アクセス許可の境界 – アイデンティティベースのポリシーで IAM エンティティに付与することのできるアクセス許可の数の上限を設定します。詳細については、「IAM ユーザーガイド」の [IAM エンティティのアクセス許可境界](#) を参照してください。
- サービスコントロールポリシー (SCP) - AWS Organizations内の組織または組織単位の最大のアクセス許可を指定します。詳細については、「AWS Organizations ユーザーガイド」の [サービスコントロールポリシー](#) を参照してください。
- リソースコントロールポリシー (RCP) – は、アカウント内のリソースで利用できる最大数のアクセス許可を定義します。詳細については、「AWS Organizations ユーザーガイド」の [リソースコントロールポリシー \(RCP\)](#) を参照してください。
- セッションポリシー – ロールまたはフェデレーションユーザーの一時セッションを作成する際にパラメータとして渡される高度なポリシーです。詳細については、「IAM ユーザーガイド」の [セッションポリシー](#) を参照してください。

複数のポリシータイプ

1つのリクエストに複数のタイプのポリシーが適用されると、結果として作成されるアクセス許可を理解するのがさらに難しくなります。が複数のポリシータイプが関与する場合にリクエストを許可するかどうか AWS を決定する方法については、「IAM ユーザーガイド」の [ポリシー評価ロジック](#) を参照してください。

How AWS DeepRacer と IAM の連携

IAM を使用して DeepRacer へのアクセスを管理する前に、DeepRacer で利用できる IAM の機能について学びます。

AWS DeepRacer で使用できる IAM 機能

IAM 機能	DeepRacer サポート
アイデンティティベースのポリシー	あり
リソースベースのポリシー	なし
ポリシーアクション	あり
ポリシーリソース	あり
ポリシー条件キー	あり
ACL	なし
ABAC (ポリシー内のタグ)	あり
一時的な認証情報	あり
プリンシパルアクセス権限	あり
サービスロール	あり
サービスリンクロール	いいえ

DeepRacer およびその他の AWS のサービスがほとんどの IAM 機能と連携する方法の概要については、IAM ユーザーガイドの[AWS 「IAM と連携する のサービス」](#)を参照してください。

DeepRacer のアイデンティティベースのポリシー

アイデンティティベースのポリシーのサポート: あり

アイデンティティベースポリシーは、IAM ユーザー、ユーザーグループ、ロールなど、アイデンティティにアタッチできる JSON 許可ポリシードキュメントです。これらのポリシーは、ユーザー

とロールが実行できるアクション、リソース、および条件をコントロールします。アイデンティティベースポリシーの作成方法については、「IAM ユーザーガイド」の「[カスタマー管理ポリシーでカスタム IAM アクセス許可を定義する](#)」を参照してください。

IAM アイデンティティベースのポリシーでは、許可または拒否するアクションとリソース、およびアクションを許可または拒否する条件を指定できます。JSON ポリシーで使用できるすべての要素について学ぶには、「IAM ユーザーガイド」の「[IAM JSON ポリシーの要素のリファレンス](#)」を参照してください。

DeepRacer のアイデンティティベースポリシーの例

DeepRacer アイデンティティベースポリシーの例を表示するには、「[AWS DeepRacer のアイデンティティベースのポリシーの例](#)」を参照してください。

DeepRacer 内のリソースベースのポリシー

リソースベースのポリシーのサポート: なし

リソースベースのポリシーは、リソースに添付する JSON ポリシードキュメントです。リソースベースのポリシーには例として、IAM ロールの信頼ポリシーや Amazon S3 バケットポリシーがあげられます。リソースベースのポリシーをサポートするサービスでは、サービス管理者はポリシーを使用して特定のリソースへのアクセスをコントロールできます。ポリシーがアタッチされているリソースの場合、指定されたプリンシパルがそのリソースに対して実行できるアクションと条件は、ポリシーによって定義されます。リソースベースのポリシーで、[プリンシパルを指定する](#)必要があります。プリンシパルには、アカウント、ユーザー、ロール、フェデレーテッドユーザー、またはを含めることができます AWS のサービス。

クロスアカウントアクセスを有効にするには、全体のアカウント、または別のアカウントの IAM エンティティを、リソースベースのポリシーのプリンシパルとして指定します。詳細については、IAM ユーザーガイドの[IAM でのクロスアカウントリソースアクセス](#)を参照してください。

DeepRacer のポリシーアクション

ポリシーアクションのサポート: あり

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

JSON ポリシーの Action 要素にはポリシー内のアクセスを許可または拒否するために使用できるアクションが記述されます。このアクションは関連付けられたオペレーションを実行するためのアクセス許可を付与するポリシーで使用されます。

DeepRacer アクションのリストを確認するには、「サービス認可リファレンス」の[AWS DeepRacer で定義されるアクション](#)」を参照してください。

DeepRacer のポリシーアクションは、アクションの前に以下のプレフィックスを使用します。

```
deepracer
```

単一のステートメントで複数のアクションを指定するには、アクションをカンマで区切ります。

```
"Action": [  
  "deepracer:action1",  
  "deepracer:action2"  
]
```

DeepRacer アイデンティティベースポリシーの例を表示するには、「[AWS DeepRacer のアイデンティティベースのポリシーの例](#)」を参照してください。

DeepRacer のポリシーリソース

ポリシーリソースのサポート: あり

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

Resource JSON ポリシー要素はアクションが適用されるオブジェクトを指定します。ベストプラクティスとして、[Amazon リソースネーム \(ARN\)](#) を使用してリソースを指定します。リソースレベルのアクセス許可をサポートしないアクションの場合は、ステートメントがすべてのリソースに適用されることを示すために、ワイルドカード (*) を使用します。

```
"Resource": "*"
```

DeepRacer リソースタイプとその ARNs [AWS DeepRacer で定義されるリソース](#)」を参照してください。各リソースの ARN を指定できるアクションについては、[AWS DeepRacer で定義されるアクション](#)」を参照してください。

DeepRacer アイデンティティベースポリシーの例を表示するには、「[AWS DeepRacer のアイデンティティベースのポリシーの例](#)」を参照してください。

DeepRacer のポリシー条件キー

サービス固有のポリシー条件キーのサポート: あり

管理者は JSON AWS ポリシーを使用して、誰が何にアクセスできるかを指定できます。つまり、どのプリンシパルがどのリソースに対してどのような条件下でアクションを実行できるかということです。

Condition 要素は、定義された基準に基づいてステートメントが実行される時期を指定します。イコールや未満などの[条件演算子](#)を使用して条件式を作成して、ポリシーの条件とリクエスト内の値を一致させることができます。すべての AWS グローバル条件キーを確認するには、「IAM ユーザーガイド」の[AWS「グローバル条件コンテキストキー」](#)を参照してください。

DeepRacer 条件キーのリストを確認するには、「サービス認可リファレンス」の「IAM ユーザーガイド」の[AWS DeepRacer の条件キー](#)」を参照してください。条件キーを使用できるアクションとリソースについては、[AWS DeepRacer で定義されるアクション](#)」を参照してください。

DeepRacer アイデンティティベースポリシーの例を表示するには、「[AWS DeepRacer のアイデンティティベースのポリシーの例](#)」を参照してください。

DeepRacer のアクセスコントロールリスト (ACL)

ACL のサポート: なし

アクセスコントロールリスト (ACL) は、どのプリンシパル (アカウントメンバー、ユーザー、またはロール) がリソースにアクセスするためのアクセス許可を持つかを制御します。ACL はリソースベースのポリシーに似ていますが、JSON ポリシードキュメント形式は使用しません。

DeepRacer での属性ベースのアクセスコントロール (ABAC)

ABAC (ポリシー内のタグ) のサポート: あり

属性ベースのアクセス制御 (ABAC) は、タグと呼ばれる属性に基づいてアクセス許可を定義する認可戦略です。IAM エンティティと AWS リソースにタグをアタッチし、プリンシパルのタグがリソースのタグと一致するときにオペレーションを許可するように ABAC ポリシーを設計できます。

タグに基づいてアクセスを管理するには、`aws:ResourceTag/key-name`、`aws:RequestTag/key-name`、または `aws:TagKeys` の条件キーを使用して、ポリシーの [条件要素](#) でタグ情報を提供します。

サービスがすべてのリソースタイプに対して 3 つの条件キーすべてをサポートする場合、そのサービスの値はありです。サービスが一部のリソースタイプに対してのみ 3 つの条件キーのすべてをサポートする場合、値は「部分的」になります。

ABAC の詳細については、「IAM ユーザーガイド」の「[ABAC 認可でアクセス許可を定義する](#)」を参照してください。ABAC をセットアップする手順を説明するチュートリアルについては、「IAM ユーザーガイド」の「[属性ベースのアクセスコントロール \(ABAC\) を使用する](#)」を参照してください。

DeepRacer での一時的な認証情報の使用

一時的な認証情報のサポート: あり

一時的な認証情報は、AWS リソースへの短期的なアクセスを提供し、フェデレーションまたはスイッチロールの使用時に自動的に作成されます。長期的なアクセスキーを使用する代わりに、一時的な認証情報を動的に生成 AWS することをお勧めします。詳細については、「IAM ユーザーガイド」の「[IAM の一時的な認証情報](#)」および「[AWS のサービスと IAM との連携](#)」を参照してください。

DeepRacer のクロスサービスプリンシパル許可

転送アクセスセッション (FAS) のサポート: あり

転送アクセスセッション (FAS) は、 を呼び出すプリンシパルのアクセス許可と AWS のサービス、ダウンストリームサービス AWS のサービス へのリクエストをリクエストする を使用します。FAS リクエストを行う際のポリシーの詳細については、「[転送アクセスセッション](#)」を参照してください。

DeepRacer のサービスロール

サービスロールのサポート: あり

サービスロールとは、サービスがユーザーに代わってアクションを実行するために引き受ける [IAM ロール](#) です。IAM 管理者は、IAM 内からサービスロールを作成、変更、削除できます。詳細については、IAM ユーザーガイドの [AWS のサービスに許可を委任するロールを作成する](#) を参照してください。

⚠ Warning

サービスロールの許可を変更すると、DeepRacer の機能が破損する可能性があります。DeepRacer が指示する場合以外は、サービスロールを編集しないでください。

DeepRacer のサービスリンクロール

サービスにリンクされたロールのサポート: なし

サービスにリンクされたロールは、にリンクされたサービスロールの一種です AWS のサービス。サービスは、ユーザーに代わってアクションを実行するロールを引き受けることができます。サービスにリンクされたロールは に表示され AWS アカウント、サービスによって所有されます。IAM 管理者は、サービスにリンクされたロールのアクセス許可を表示できますが、編集することはできません。

サービスにリンクされたロールの作成または管理の詳細については、「[IAM と提携するAWS のサービス](#)」を参照してください。表の「サービスリンクロール」列に Yes と記載されたサービスを見つけます。サービスにリンクされたロールに関するドキュメントをサービスで表示するには、[はい] リンクを選択します。

AWS DeepRacer のアイデンティティベースのポリシーの例

デフォルトでは、IAM ユーザーおよびロールには、DeepRacer リソースを作成または変更するアクセス許可はありません。IAM 管理者は、リソースで必要なアクションを実行するための権限をユーザーに付与する IAM ポリシーを作成できます。

これらのサンプルの JSON ポリシードキュメントを使用して IAM アイデンティティベースのポリシーを作成する方法については、「IAM ユーザーガイド」の「[IAM ポリシーを作成する \(コンソール\)](#)」を参照してください。

各リソースタイプの ARN の形式など、DeepRacer で定義されるアクションとリソースタイプの詳細については、「サービス認可リファレンス」の[AWS DeepRacer のアクション、リソース、および条件キー](#)」を参照してください。ARNs

トピック

- [ポリシーに関するベストプラクティス](#)
- [DeepRacer コンソールを使用する](#)

- [自分の権限の表示をユーザーに許可する](#)

ポリシーに関するベストプラクティス

ID ベースのポリシーは、ユーザーのアカウントで誰かが DeepRacer リソースを作成、アクセス、または削除できるどうかを決定します。これらのアクションでは、AWS アカウントに費用が発生する場合があります。アイデンティティベースポリシーを作成したり編集したりする際には、以下のガイドラインと推奨事項に従ってください:

- AWS 管理ポリシーを開始し、最小特権のアクセス許可に移行する – ユーザーとワークロードにアクセス許可の付与を開始するには、多くの一般的なユースケースにアクセス許可を付与するAWS 管理ポリシーを使用します。これらはで使用できます AWS アカウント。ユースケースに固有の AWS カスタマー管理ポリシーを定義することで、アクセス許可をさらに減らすことをお勧めします。詳細については、IAM ユーザーガイドの [AWS マネージドポリシー](#) または [ジョブ機能のAWS マネージドポリシー](#) を参照してください。
- 最小特権を適用する – IAM ポリシーでアクセス許可を設定する場合は、タスクの実行に必要な許可のみを付与します。これを行うには、特定の条件下で特定のリソースに対して実行できるアクションを定義します。これは、最小特権アクセス許可とも呼ばれています。IAM を使用して許可を適用する方法の詳細については、IAM ユーザーガイドの [IAM でのポリシーとアクセス許可](#) を参照してください。
- IAM ポリシーで条件を使用してアクセスをさらに制限する - ポリシーに条件を追加して、アクションやリソースへのアクセスを制限できます。たとえば、ポリシー条件を記述して、すべてのリクエストを SSL を使用して送信するように指定できます。条件を使用して、サービスアクションがなどの特定のを通じて使用されている場合に AWS のサービス、サービスアクションへのアクセスを許可することもできます CloudFormation。詳細については、IAM ユーザーガイドの [IAM JSON ポリシー要素:条件](#) を参照してください。
- IAM アクセスアナライザーを使用して IAM ポリシーを検証し、安全で機能的な権限を確保する - IAM アクセスアナライザーは、新規および既存のポリシーを検証して、ポリシーが IAM ポリシー言語 (JSON) および IAM のベストプラクティスに準拠するようにします。IAM アクセスアナライザーは 100 を超えるポリシーチェックと実用的な推奨事項を提供し、安全で機能的なポリシーの作成をサポートします。詳細については、IAM ユーザーガイドの [IAM Access Analyzer でポリシーを検証する](#) を参照してください。
- 多要素認証 (MFA) が必要 – で IAM ユーザーまたはルートユーザーを必要とするシナリオがある場合は AWS アカウント、セキュリティを強化するために MFA を有効にします。API オペレーションが呼び出されるときに MFA を必須にするには、ポリシーに MFA 条件を追加します。詳細については、IAM ユーザーガイドの [MFA を使用した安全な API アクセス](#) を参照してください。

IAM でのベストプラクティスの詳細については、IAM ユーザーガイドの [IAM でのセキュリティのベストプラクティス](#) を参照してください。

DeepRacer コンソールを使用する

AWS DeepRacer コンソールにアクセスするには、最小限のアクセス許可のセットが必要です。これらのアクセス許可により、AWS アカウントの DeepRacer リソースの詳細をリストおよび表示できます。最小限必要な許可よりも制限が厳しいアイデンティティベースのポリシーを作成すると、そのポリシーを持つエンティティ (ユーザーまたはロール) に対してコンソールが意図したとおりに機能しません。

AWS CLI または AWS API のみ呼び出すユーザーには、最小限のコンソールアクセス許可を付与する必要はありません。代わりに、実行しようとしている API オペレーションに一致するアクションのみへのアクセスが許可されます。

ユーザーとロールが DeepRacer コンソールを引き続き使用できるようにするには、エンティティに DeepRacer ConsoleAccess または ReadOnly AWS 管理ポリシーもアタッチします。詳細については、「IAM ユーザーガイド」の「[ユーザーへの許可の追加](#)」を参照してください。

自分の権限の表示をユーザーに許可する

この例では、ユーザーアイデンティティにアタッチされたインラインおよびマネージドポリシーの表示を IAM ユーザーに許可するポリシーの作成方法を示します。このポリシーには、コンソールで、または AWS CLI または AWS API を使用してプログラムでこのアクションを実行するアクセス許可が含まれています。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsWithUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
      ],
      "Resource": ["arn:aws:iam::*:user/${aws:username}"]
    }
  ],
}
```

```
{
  "Sid": "NavigateInConsole",
  "Effect": "Allow",
  "Action": [
    "iam:GetGroupPolicy",
    "iam:GetPolicyVersion",
    "iam:GetPolicy",
    "iam:ListAttachedGroupPolicies",
    "iam:ListGroupPolicies",
    "iam:ListPolicyVersions",
    "iam:ListPolicies",
    "iam:ListUsers"
  ],
  "Resource": "*"
}
]
```

AWS AWS DeepRacer の マネージドポリシー

AWS 管理ポリシーは、によって作成および管理されるスタンドアロンポリシーです AWS。AWS 管理ポリシーは、ユーザー、グループ、ロールにアクセス許可の割り当てを開始できるように、多くの一般的なユースケースにアクセス許可を付与するように設計されています。

AWS 管理ポリシーは、すべての AWS お客様が使用できるため、特定のユースケースに対して最小特権のアクセス許可を付与しない場合があることに注意してください。ユースケースに固有の[カスタマー管理ポリシー](#)を定義して、アクセス許可を絞り込むことをお勧めします。

AWS 管理ポリシーで定義されているアクセス許可は変更できません。が AWS マネージドポリシーで定義されたアクセス許可 AWS を更新すると、ポリシーがアタッチされているすべてのプリンシパル ID (ユーザー、グループ、ロール) に影響します。AWS は、新しい が起動されるか、新しい API オペレーション AWS のサービス が既存のサービスで使用できるようになったときに、AWS マネージドポリシーを更新する可能性が高くなります。

詳細については、「IAM ユーザーガイド」の「[AWS マネージドポリシー](#)」を参照してください。

以下の AWS 管理ポリシーは、AWS DeepRacer マルチユーザーモードを使用して AWS、アカウント内の複数の参加者をスポンサーすることに固有のものです。

- AWSDeepRacerAccountAdminAccess マルチユーザーアカウント管理者に必要な AWS DeepRacer アクセス権限を付与します。

- `AWSDeepRacerDefaultMultiUserAccess` AWS DeepRacer コンソールを使用するために必要な AWS DeepRacer アクセス権限を付与します。

トピック

- [AWS DeepRacer Account Admin Access AWS DeepRacer 管理者の管理ポリシー](#)
- [AWS DeepRacer Default Multi User Access AWS DeepRacer マルチユーザーレーサー向けの管理ポリシー](#)
- [AWS DeepRacer AWS 管理ポリシーの更新](#)

AWS DeepRacer Account Admin Access AWS DeepRacer 管理者の管理ポリシー

複数のプロフィールが AWS DeepRacer で AWS アカウント ID と請求情報を使用できるようにするには、`AWSDeepRacerAccountAdminAccess` ポリシーをアタッチします。

他のレーサーのスポンサー時に使用する IAM アイデンティティに `AWSDeepRacerAccountAdminAccess` ポリシーをアタッチできます。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DeepRacerAdminAccessStatement",
      "Effect": "Allow",
      "Action": [
        "deepracer:*"
      ],
      "Resource": [
        "*"
      ],
      "Condition": {
        "Null": {
          "deepracer:UserToken": "true"
        }
      }
    }
  ]
}
```

```
    }
  }
]
}
```

AWS DeepRacer Default Multi User Access AWS DeepRacer マルチユーザーレーサー向けの管理ポリシー

ポリシー `AWSDeepRacerDefaultMultiUserAccess` は、AWS DeepRacer レーサーに、マルチユーザーアカウントの管理者アクションを除くすべての AWS DeepRacer アクションへのアクセスを許可します。

アカウント内でスポンサーしたい参加者の IAM アイデンティティに `AWSDeepRacerDefaultMultiUserAccess` ポリシーをアタッチできます。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "deepracer:Add*",
        "deepracer:Remove*",
        "deepracer:Create*",
        "deepracer:Perform*",
        "deepracer:Clone*",
        "deepracer:Get*",
        "deepracer:List*",
        "deepracer>Edit*",
        "deepracer:Start*",
        "deepracer:Set*",
        "deepracer:Update*",
        "deepracer>Delete*",
        "deepracer:Stop*",
        "deepracer:Import*",
        "deepracer:Tag*",
        "deepracer:Untag*"
      ],
      "Resource": [
```

```
    "*"
  ],
  "Condition": {
    "Null": {
      "deepracer:UserToken": "false"
    },
    "Bool": {
      "deepracer:MultiUser": "true"
    }
  }
},
{
  "Effect": "Allow",
  "Action": [
    "deepracer:GetAccountConfig",
    "deepracer:GetTrack",
    "deepracer:ListTracks",
    "deepracer:TestRewardFunction"
  ],
  "Resource": [
    "*"
  ]
},
{
  "Effect": "Deny",
  "Action": [
    "deepracer:Admin*"
  ],
  "Resource": [
    "*"
  ]
}
]
```

AWS DeepRacer AWS 管理ポリシーの更新

このサービスがこれらの変更の追跡を開始してからの AWS DeepRacer の AWS マネージドポリシーの更新に関する詳細を表示します。このページの変更に関する自動通知については、[AWS DeepRacer ドキュメントの履歴] ページの [RSS フィード] を購読してください。

変更	説明	日付
AWSDeepRacerAccountAdminAccess と AWSDeepRacerDefaultMultiUserAccess ポリシーが追加されました。	追加された新しい管理ポリシーによって、マルチユーザーモードを使用して1つのAWS DeepRacer アカウントで複数の参加者をスポンサーできるようになりました。	2021年10月26日
AWS DeepRacer がポリシー変更の追跡を開始しました。	AWS DeepRacer が、AWS 管理ポリシーに対する変更の追跡を開始しました。	2021年10月26日

サービス間での不分別な代理処理の防止

混乱した代理問題は、アクションを実行するためのアクセス許可を持たないエンティティが、より特権のあるエンティティにアクションの実行を強制できてしまう場合に生じる、セキュリティ上の問題です。では AWS、サービス間のなりすましにより、混乱した代理問題が発生する可能性があります。サービス間でのなりすましは、1つのサービス(呼び出し元サービス)が、別のサービス(呼び出し対象サービス)を呼び出すときに発生する可能性があります。呼び出し元サービスは、本来ならアクセスすることが許可されるべきではない方法でその許可を使用して、別のお客様のリソースに対する処理を実行するように操作される場合があります。これを防ぐため、AWS では、アカウントのリソースへのアクセス権が付与されたサービスプリンシパルで、すべてのサービスのデータを保護するために役立つツールを提供しています。

リソースポリシー内では [aws:SourceArn](#) および [aws:SourceAccount](#) のグローバル条件コンテキストキーを使用して、AWSDeepRacerLong が別のサービスに付与する、リソースへのアクセス許可を制限することをお勧めします。両方のグローバル条件コンテキストキーを同じポリシーステートメントで使用する場合は、aws:SourceAccount 値と、aws:SourceArn 値に含まれるアカウントが、同じアカウント ID を示している必要があります。

aws:SourceArn の値が s3:::your-bucket-name である必要があります。

混乱した代理問題から保護するための最も効果的な方法は、リソースの完全な ARN を指定して aws:SourceArn グローバル条件コンテキストキーを使用することです。リソースの完全な ARN が不明な場合や、複数のリソースを指定する場合は、aws:SourceArn グローバル

コンテキスト条件キーを使用して、ARN の未知部分をワイルドカード (*) で表します。例えば、arn:aws:*servicename*::*123456789012*:*。

次の例では、AWSDeepRacer で aws:SourceArn および aws:SourceAccount グローバル条件コンテキストキーを使用して、「Confused Deputy Problem (混乱した使節の問題)」を回避する方法を示します。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1586917903457",
      "Effect": "Allow",
      "Principal": {
        "Service": "deepracer.amazonaws.com"
      },
      "Action": [
        "s3:GetObjectAcl",
        "s3:GetObject",
        "s3:PutObject",
        "s3:PutObjectAcl"
      ],
      "Resource": [
        "arn:aws:s3:::your-bucket-name",
        "arn:aws:s3:::your-bucket-name/*"
      ],
      "Condition": {
        "StringEquals": {
          "aws:SourceArn": "arn:${Partition}:deepracer:${us-east-1}:${Account}:model/reinforcement_learning/${ResourceId}"
        }
      }
    }
  ]
}
```

このバケットにカスタム AWS Key Management Service (KMS) リソースを使用する場合は、AWS KMS リソースポリシーを含めます。

トラブルシューティング AWS DeepRacer アイデンティティとアクセス

次の情報は、DeepRacer と IAM の使用に伴って発生する可能性がある一般的な問題の診断や修復に役立ちます。

トピック

- [DeepRacer マルチユーザーアカウントモードで認可エラーが発生します。](#)
- [DeepRacer でアクションを実行する権限がありません。](#)
- [iam:PassRole を実行する権限がない](#)
- [アクセスキーを表示したい](#)
- [管理者として DeepRacer へのアクセスを他のユーザーに許可したいです。](#)
- [自分の 以外のユーザーに DeepRacer リソース AWS アカウント へのアクセスを許可したい](#)

DeepRacer マルチユーザーアカウントモードで認可エラーが発生します。

[AWSDeepRacerAccountAdminAccess](#) ポリシーを持つ管理者の場合、セッションに関連付けられたユーザートークンがあると、認可エラーが発生することがあります。管理者は、セッションに関連付けられたユーザートークンを持つべきではありません。これを解決するには、cookie をクリアしてください。

アカウントがマルチユーザーモードにあり、[AWSDeepRacerDefaultMultiUserAccess](#) ポリシーを持つレーサーである場合、ポリシーに関連付けられたユーザートークンがない場合、認可エラーが発生することがあります。これを解決するには、AWS DeepRacer の使用を続行する前に、AWS Player プロファイルに対して認証する必要があります。

アカウントがシングルユーザーモードで、[AWSDeepRacerDefaultMultiUserAccess](#) ポリシーを持つレーサーである場合は、認可エラーが発生する可能性があります。これを解決するには、シングルユーザーモードでは、[AWSDeepRacerDefaultMultiUserAccess](#) ポリシーを持つユーザーは AWS DeepRacer を使用できないため、AWS アカウント管理者に確認してください。

DeepRacer でアクションを実行する権限がありません。

でアクションを実行する権限がないと AWS マネジメントコンソール 通知された場合は、管理者に連絡してサポートを依頼する必要があります。管理者とは、ユーザーにユーザー名とパスワードを提供した人です。

以下のエラー例は、mateojackson IAM ユーザーがコンソールを使用して架空の *my-example-widget* リソースに関する詳細情報を表示しようとしているが、架空の `deepracer:GetWidget` アクセス許可がないという場合に発生します。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
deepracer:GetWidget on resource: my-example-widget
```

この場合、Mateo は、`deepracer:GetWidget` アクションを使用して *my-example-widget* リソースにアクセスできるように、ポリシーの更新を管理者に依頼します。

iam:PassRole を実行する権限がない

`iam:PassRole` アクションを実行する権限がないというエラーが表示された場合は、ポリシーを更新して DeepRacer にロールを渡すことができるようにする必要があります。

一部の AWS のサービスでは、新しいサービスロールまたはサービスにリンクされたロールを作成する代わりに、そのサービスに既存のロールを渡すことができます。そのためには、サービスにロールを渡すアクセス許可が必要です。

以下の例のエラーは、marymajor という IAM ユーザーがコンソールを使用して DeepRacer でアクションを実行しようする場合に発生します。ただし、このアクションをサービスが実行するには、サービスロールから付与されたアクセス許可が必要です。Mary には、ロールをサービスに渡すアクセス許可がありません。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

この場合、Mary のポリシーを更新してメアリーに `iam:PassRole` アクションの実行を許可する必要があります。

サポートが必要な場合は、AWS 管理者にお問い合わせください。サインイン資格情報を提供した担当者が管理者です。

アクセスキーを表示したい

IAM ユーザーアクセスキーを作成した後は、いつでもアクセスキー ID を表示できます。ただし、シークレットアクセスキーを再表示することはできません。シークレットアクセスキーを紛失した場合は、新しいアクセスキーペアを作成する必要があります。

アクセスキーは、アクセスキー ID (例: AKIAIOSFODNN7EXAMPLE) とシークレットアクセスキー (例: wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY) の 2 つで構成されています。ユーザー名とパ

スワードと同様に、リクエストを認証するために、アクセスキー ID とシークレットアクセスキーの両方を使用する必要があります。ユーザー名とパスワードと同様に、アクセスキーは安全に管理してください。

⚠ Important

正規のユーザー ID を確認するためであっても、アクセスキーを第三者に提供しないでください。これにより、への永続的なアクセス権をユーザーに付与できます AWS アカウント。

アクセスキーペアを作成する場合、アクセスキー ID とシークレットアクセスキーを安全な場所に保存するように求めるプロンプトが表示されます。このシークレットアクセスキーは、作成時にのみ使用できます。シークレットアクセスキーを紛失した場合、IAM ユーザーに新規アクセスキーを追加する必要があります。アクセスキーは最大 2 つまで持つことができます。既に 2 つある場合は、新規キーペアを作成する前に、いずれかを削除する必要があります。手順を表示するには、IAM ユーザーガイドの「[アクセスキーの管理](#)」を参照してください。

管理者として DeepRacer へのアクセスを他のユーザーに許可したいです。

DeepRacer へのアクセスを他のユーザーに許可するには、アクセスを必要とするユーザーまたはアプリケーションにアクセス許可を付与する必要があります。AWS IAM Identity Center を使用してユーザーとアプリケーションを管理する場合は、アクセスレベルを定義するアクセス許可セットをユーザーまたはグループに割り当てます。アクセス許可セットは、ユーザーまたはアプリケーションに関連付けられている IAM ロールに自動的に IAM ポリシーを作成して割り当てます。詳細については、「AWS IAM Identity Center ユーザーガイド」の「[アクセス許可セット](#)」を参照してください。

IAM アイデンティティセンターを使用していない場合は、アクセスを必要としているユーザーまたはアプリケーションの IAM エンティティ (ユーザーまたはロール) を作成する必要があります。次に、DeepRacer の適切なアクセス許可を付与するポリシーを、そのエンティティにアタッチする必要があります。アクセス許可が付与されたら、ユーザーまたはアプリケーション開発者に認証情報を提供します。これらの認証情報を使用して AWS にアクセスします。IAM ユーザー、グループ、ポリシー、アクセス許可の作成の詳細については、「IAM ユーザーガイド」の「[IAM アイデンティティ](#)」と「[IAM のポリシーとアクセス許可](#)」を参照してください。

自分の以外のユーザーに DeepRacer リソース AWS アカウント へのアクセスを許可したい

他のアカウントのユーザーや組織外の人が、リソースにアクセスするために使用できるロールを作成できます。ロールの引き受けを委託するユーザーを指定できます。リソースベースのポリシーまた

はアクセスコントロールリスト (ACL) をサポートするサービスの場合、それらのポリシーを使用して、リソースへのアクセスを付与できます。

詳細については、以下を参照してください:

- DeepRacer がこれらの機能をサポートしているかどうかを確認するには、「[How AWS DeepRacer と IAM の連携](#)」を参照してください。
- 所有 AWS アカウントしているのリソースへのアクセスを提供する方法については、IAM ユーザーガイドの「[所有 AWS アカウントしている別の IAM ユーザーへのアクセスを提供する](#)」を参照してください。
- リソースへのアクセスをサードパーティーに提供する方法については AWS アカウント、IAM ユーザーガイドの「[サードパーティー AWS アカウント が所有する へのアクセスを提供する](#)」を参照してください。
- ID フェデレーションを介してアクセスを提供する方法については、IAM ユーザーガイドの [外部で認証されたユーザー \(ID フェデレーション\) へのアクセスの許可](#) を参照してください。
- クロスアカウントアクセスにおけるロールとリソースベースのポリシーの使用法の違いについては、「IAM ユーザーガイド」の「[IAM でのクロスアカウントのリソースへのアクセス](#)」を参照してください。

Tagging

タグは、AWS リソース AWS に割り当てるカスタム属性ラベルです。各 AWS タグには 2 つの部分があります。

- タグキー (例: [companyname、costcenter、environment、project、またはsecret])。タグキーでは大文字と小文字が区別されます。
- タグ値として知られるオプションのフィールドです。タグ値を省略すると、空の文字列を使用した場合と同じになります。タグキーと同様に、タグ値では大文字と小文字が区別されます。

これらを合わせて、キーと値のペアと呼ばれます。

AWS DeepRacer サービス内では、車や RL モデル、コミュニティレースのリーダーボードなどにタグを割り当てることができます。タグ付けをサポートするこれらの AWS リソースと他のリソースにタグを付けて、リソースが関連していることを示します。タグを使うことで、モデルやリーダーボードの識別・整理に加え、コスト配分を把握したり、IAM ポリシーで誰がリソースを閲覧・操作できるかを管理することもできます。AWS DeepRacer コンソールまたは AWS CLI を使用して、タグを追加、管理、削除します。

タグの使用の詳細については、「[タグ付けのベストプラクティス](#)」ホワイトペーパーを参照してください。

タグでコスト分配を把握

AWS Cost Explorer とコスト・使用状況レポートは、AWSのコストをタグ別に分類する機能をサポートします。AWS のコストを組織の代表的な財務報告カテゴリーに関連付けるために、cost center、businessunit、またはprojectなどのビジネスタグを使用できます。しかしながら、コスト配分レポートは、特定のアプリケーション、環境、コンプライアンス プログラムなどの技術的またはセキュリティ上のカテゴリーとコストを簡単に関連付けることができるタグを含めることができます。請求情報とコスト管理コンソールの [Cost Allocation Tags] マネージャーにアクセスできるのは、組織の管理アカウントおよび組織のメンバーではない単一のアカウントのみです。タグを使ったコスト配分把握の詳細については、[\[ユーザー定義のコスト配分タグ\]](#) を参照してください。

タグでアクセス管理

IAM ユーザーとロールをタグ付けして、モデルやコミュニティレースのリーダーボードへのアクセスを管理することもできます。IAM ユーザーとロールをタグ付けする方法については、[\[IAM ユー](#)

[ザーとロールをタグ付けする](#)]を参照してください。プリンシパルタグを持つ IAM ロールが一致するタグを持つリソースにアクセスできるようにするポリシーを作成およびテストするためのチュートリアルを表示するには、[「IAM チュートリアル: タグに基づいて AWS リソースにアクセスするためのアクセス許可を定義する」](#)を参照してください。タグを使用してタグ付けをサポートする AWS リソースへのアクセスを制御する方法の詳細については、[AWS 「リソースタグを使用したリソースへのアクセスの制御」](#)を参照してください。

トピック

- [新規リソースへタグの追加、表示、編集を行う。](#)
- [既存のリソースにタグの追加、表示、編集を行う](#)

新規リソースへタグの追加、表示、編集を行う。

新車、RL モデル、コミュニティレースのリーダーボードなどにタグを追加することで、これらのリソースの識別、整理、コスト配分の把握、アクセスの管理などが可能になります。モデルやリーダーボードに 1 つまたは複数のタグ (キーバリューのペア) を追加します。各タグキーは、各リソースに対しそれぞれ一意である必要があり、各タグキーに設定できる値は 1 つのみですが、1 つのリソースは最大 50 個のタグを持つことができます。

AWS DeepRacer コンソールで 1 つずつリソースを作成してタグを適用するか、[タグエディタ](#)を使用して複数のリソースを一度に追加、編集、削除します。

Important

RL モデルやコミュニティレースのリーダーボードのタグを編集すると、それらのリソースへのアクセスに影響を与えます。タグの名前 (キー) や値を編集する前に、タグのキーや値を使用してリソースへのアクセスを制御する可能性のある IAM ポリシーがあれば、必ず確認してください。

新規 RL モデルにタグを追加、表示、編集するには

AWS DeepRacer コンソールを使用して、新規 RL モデルにタグを追加、表示、および編集します。

1. [モデル] ページで、[モデル作成] を選択します。
2. [モデルの作成] ページで [トレーニングの詳細] を入力した後、[タグ] 見出しを展開します。
3. [タグ] の見出しで、[新しいタグを追加] を選択します。

4. [キー] に、タグの名前を入力します。[値] では、オプションでタグに値を追加できます。命名タグの詳細については、「[タグ付けのベストプラクティス](#)」ホワイトペーパーに記載された「命名タグとリソースに関するベストプラクティス」のトピックを参照してください。
5. (オプション) 別のタグを追加するには、[タグを追加する] を再度選択します。
6. (オプション) 個々のキーまたは値を削除するには、その横にある [X] を選択します。
7. (オプション) キーと値のペアを削除するには、[削除] を選択します。
8. タグの追加が完了したら、[環境シミュレーション] を選択し、[次へ] を選択します。

新しいモデルをトレーニング用にタグ付けして送信した後、ページの下部にある [タグ] 見出しでトレーニングの最中や後、評価中にタグの管理をすることができます。

1. [タグの管理] を選択します。
2. [タグの管理] ポップアップボックスでは、タグの横にある [削除] ボタンを選択し作成したタグを削除、または [新規タグを追加する] を選択し、新しいタグを追加します。
3. 新しいタグを追加する場合は、[キー] に、タグの名前を入力します。[値] では、任意でタグに値を追加できます。命名タグの詳細については、「[タグ付けのベストプラクティス](#)」ホワイトペーパーに記載された「命名タグとリソースに関するベストプラクティス」のトピックを参照してください。
4. タグの削除を完了したら、[送信] を選択します。

新しいコミュニティ レースリーダーボードのタグを追加、表示、編集するには

AWS DeepRacer コンソールを使用して、新しいコミュニティ レースリーダーボードにタグを追加、表示、編集します。

1. [コミュニティレース] で、[レースを作成] を選択します。
2. [レースの詳細] ページで [タグ] の見出しを展開します。
3. [タグ] の見出しで、[新しいタグを追加] を選択します。
4. [キー] に、タグの名前を入力します。[値] では、オプションでタグに値を追加できます。命名タグの詳細については、「[タグ付けのベストプラクティス](#)」ホワイトペーパーに記載された「命名タグとリソースに関するベストプラクティス」のトピックを参照してください。
5. (オプション) 別のタグを追加するには、[タグを追加する] を再度選択します。
6. (オプション) 個々のキーまたは値を削除するには、その横にある [X] を選択します。
7. (オプション) キーと値のペアを削除するには、[削除] を選択します。

8. タグの追加が完了したら、[環境シミュレーション] を選択し、[次へ] を選択します。

既存のリソースにタグの追加、表示、編集を行う

既存の AWS DeepRacer RL モデルまたはコミュニティ レースリーダーボードにタグを追加することで、これらのリソースの識別、整理、コスト配分の把握、アクセスの管理などが可能になります。モデルやリーダーボードに 1 つまたは複数のタグ (キーと値のペア) を追加します。各タグキーは、各リソースに対しそれぞれ一意である必要があり、各タグキーに設定できる値は 1 つのみですが、1 つのリソースは最大 50 個のタグを持つことができます。

AWS DeepRacer コンソールで 1 つずつリソースを作成してタグを適用するか、[タグエディタ](#)を使用して複数のリソースを一度に追加、編集、削除します。

Important

RL モデルやコミュニティレースのリーダーボードのタグを編集すると、それらのリソースへのアクセスに影響を与えます。タグの名前 (キー) または値を編集する前に、タグのキーまたは値を使用してリソースへのアクセスを制御する可能性のある IAM ポリシーがあれば、必ず確認してください。

既存の RL モデルのタグを追加、表示、編集するには

AWS DeepRacer コンソールを使用して、既存の RL モデルのタグを追加、表示、編集できます。

1. [モデル] のリストの中からモデルを名前を選択します。
2. [アクション] を選択します。
3. ドロップダウンリストから [タグの管理] を選択します。
4. [タグの管理] のポップアップボックス内でタグの表示、追加、削除を行うことができます。
 - a. タグを追加するには、[新しいタグの追加] を選択します。[キー] に、タグの名前を入力します。[値] では、オプションでタグに値を追加できます。タグの名前付けに関する詳細は、[\[タグ付け ベストプラクティス\]](#) ホワイトペーパー内、「タグとリソースの名前付けに関するベストプラクティス」のトピックを参照してください。
 - b. 別のタグを追加するには、もう一度 [新しいタグを追加] を選択します。
 - c. 個別のキーまたは値を削除するには、その横にある [X] を選択します。
 - d. キーと値のペアを削除するには、[削除] を選択します。

5. タグの表示、追加、削除が完了したら、[送信] を選択します。

既存のコミュニティ レースリーダーボードにタグを追加、表示、編集するには

1. [コミュニティレース] で、[レースの管理] を選択します。
2. [レースの管理] のページで、レースを選択します
3. [アクション] を選択
4. ドロップダウンリストから [タグの管理] を選択します。
5. [タグの管理] のポップアップボックス内でタグの表示、追加、削除を行うことができます。
 - a. タグを追加するには、[新しいタグの追加] を選択します。[キー] に、タグの名前を入力します。[値] では、オプションでタグに値を追加できます。タグの名前付けに関する詳細は、[\[タグ付け ベストプラクティス\]](#) ホワイトペーパー内、「タグとリソースの名前付けに関するベストプラクティス」のトピックを参照してください。
 - b. 別のタグを追加するには、もう一度 [新しいタグを追加] を選択します。
 - c. 個別のキーまたは値を削除するには、その横にある [X] を選択します。
 - d. キーと値のペアを削除するには、[削除] を選択します。
6. タグの表示、追加、削除が完了したら、[送信] を選択します。

AWS DeepRacerの一般的な問題に関するトラブルシューティング

ここでは、よくある質問と最新のバグ修正に関するトラブルシューティングのヒントを紹介します。

トピック

- [AWS DeepRacer LIVE の一般的な問題を解決する方法](#)
- [コンピュータと車両の間で、デバイスコンソールと USB を接続できないのはなぜですか？](#)
- [AWS DeepRacer コンピューティングモジュールの電源を、バッテリーからコンセントに切り替える方法](#)
- [USB フラッシュドライブを使用して AWS DeepRacer を Wi-Fi ネットワークに接続する方法](#)
- [AWS DeepRacer ドライブモジュールのバッテリーを充電する方法](#)
- [AWS DeepRacer コンピューティングモジュールのバッテリーを充電する方法](#)
- [バッテリーは充電されていますが、AWS DeepRacer 車両が動きません。](#)
- [AWS DeepRacer 車両バッテリーのロックアウトのトラブルシューティング](#)
- [LiDAR センサーの取り付け時に Dell バッテリーコネクタケーブルを巻きつける方法](#)
- [車両の Wi-Fi 接続を維持する方法](#)
- [AWS DeepRacer デバイスの MAC アドレスを取得する方法](#)
- [AWS DeepRacer デバイスコンソールのデフォルトパスワードを回復する方法](#)
- [AWS DeepRacer デバイスを手動で更新するには](#)
- [AWS DeepRacer の一般的な操作問題の診断および解決方法](#)

AWS DeepRacer LIVE の一般的な問題を解決する方法

LIVE レースのページでレースのビデオが見れません。

- 仮想プライベートネットワーク (VPN) を使用している場合は、レースイベント中には接続が切断されていることを確認します。
- デバイスが広告ブロッカーを実行している場合は、レースイベント中には広告ブロッカーが切断されていることを確認します。
- ホームネットワークが広告ブロッカーを実行している場合は、レースイベント中には広告ブロッカーが切断されていることを確認します。

レースキューのレーサーの名前は赤です。

[LIVE: <Your Race Name>] ページのカミングアップのセクションでレーサーの名前が赤でハイライトされている時は、レーサーのモデルの設定に何か問題があったことを意味します。

- レース主催者で、レーサーのモデルの設定を削除する場合は、[LIVE: <Your Race Name>] ページのカミングアップの欄で [編集] を選択し、該当するレーサーの名前が入っている行の [X] を選択します。次に [保存] を選択します。キューの順序を変更する際には、[the section called “LIVE レースで走行する”](#) のステップ 11 を参照します。

The screenshot shows the AWS DeepRacer LIVE interface for a race named "TESTLIVERACE". The interface is divided into several sections:

- Header:** "LIVE: TESTLIVERACE" with a "View leaderboard" link.
- Race Details:** "Start time: 2:00 PM local, July 2", "Time trial race", "Cumulo Turnpike track", "Best lap time", "Unlimited resets".
- LEADERBOARD:** A table with 8 rows, each containing a rank (#1 to #8) and a placeholder for a racer's name and time.
- COMING UP:** A section with a toggle for "Model entries closed" (currently off) and a "Save" button. Below this is a table of racers:

Racer up next	Time	Action
racer2		Launch
racer1	2:09 PM	X
racer	2:14 PM	X
- Race organizer control panel:** Includes "Open broadcast mode" and "Declare winner!" buttons.
- Race simulator:** Shows "Status: Running (Updated: 4 seconds ago)" and a "Reset simulator" button.
- Current ranked submissions:** "0 Leaderboard can be cleared when no submissions are in progress." with a "Clear leaderboard ranking" button.
- Launch live racing simulator:** Three steps: "Ready" (20-25 minutes), "Set" (5-10 minutes), and "Go" (Instant).

- お客様がレーサーの場合は、モデルをレースに再提出します。ヘルプが必要な場合は [the section called “LIVE レースで走行する”](#) に移動し、[LIVE レースに参加する] を選択します。

LIVE レースを起動しているため、レーサーを立ち上げられません。

- [LIVE: <Your Race Name>] ページ内の [ライブレースシミュレーターを起動する] のセクションで [シミュレータの起動] が選択されていることを確認します。その他のヘルプについては、「[the section called “LIVE レースで走行する”](#)」のステップ 2」を参照します。

The screenshot displays the AWS DeepRacer interface for a live race named 'TESTLIVERACE'. The main content area features a large grey box with the text 'Welcome to TestLiveRace LIVE!' and instructions for the race organizer. Below this is the 'Race organizer control panel' which includes a 'Launch simulator' button highlighted with a red circle. To the right, the 'LEADERBOARD' section shows 8 empty slots for racers. At the bottom right, the 'COMING UP' section has a 'Model entries open' toggle set to 'on'.

- 設定を閉じるには [LIVE: <Your Race Name>] ページ内の [カミングアップ] で [モデルエントリーを開く] がオフになっていることを確認します。その他のヘルプについては、「[the section called “LIVE レースで走行する”](#)」のステップ 3」を参照します。

AWS DeepRacer > Community races > TestLiveRace > LIVE

LIVE: TESTLIVERACE

View leaderboard

Start time: 2:00 PM local, July 2
Time trial race
Cumulo Turnpike track
Best lap time
Unlimited resets

LEADERBOARD

#1	----
#2	----
#3	----
#4	----
#5	----
#6	----
#7	----
#8	----

COMING UP

Model entries open
Toggle off to edit race queue Edit

Racer up next	Time
---------------	------

Race organizer control panel

Open broadcast mode Declare winner!

Race simulator Status: Not created
Reset simulator

Current ranked submissions: 0
Leaderboard can be cleared when no submissions are in progress.
Clear leaderboard ranking

Launch live racing simulator

- Ready** Activate race simulator to run live race up to an hour before start time. 20-25 minutes Launch simulator
- Set** Take roll call. Ensure racers are ready. Edit queue by turning off model entries. 5-10 minutes
- Go!** Launch your first racer in the queue. Instant

Chrome または Firefox のブラウザ使っていますが、まだ LIVE レースを見ることができません。

- Chrome または Firefox ブラウザの最新バージョンがインストールされていることを確認します。そうでない場合は、ブラウザを最新バージョンに更新してから、再度試します。
- 仮想プライベートネットワーク (VPN) を使用している場合は、接続が切断されていることを確認します。
- デバイスが広告ブロッカーを実行している場合は、レースイベント中には広告ブロッカーが切断されていることを確認します。
- ホームネットワークが広告ブロッカーを実行している場合は、レースイベント中には広告ブロッカーが切断されていることを確認します。

- WebRTC がインターネットブラウザでオフになっている場合は、レースイベント中には WebRTC をオンにします。

コンピュータと車両の間で、デバイスコンソールと USB を接続できないのはなぜですか？

車両を初めて設定する場合は、マイクロ USB/USB ケーブル (USB は USB-A と呼ばれます) を使用してコンピュータに AWS DeepRacer 車両を接続後に、デバイスコンソール (車両でホストされるデバイスウェブサーバー <https://deepracer.aws> と呼ばれます) を開くことができない場合があります。

これには複数の原因があります。通常、以下の簡単な解決策で問題を解決できます。

デバイスの USB-over-Ethernet ネットワークをアクティブにするには

1. コンピュータの Wi-Fi をオフにし、接続されているイーサネットケーブルをすべて取り外します。
2. 車両の [リセット] ボタンを押して、デバイスを再起動します。
3. コンピュータのウェブブラウザから <https://deepracer.aws> に移動して、デバイスコンソールを開きます。

上記の手順で問題が解決しない場合は、コンピュータのネットワーク設定をチェックし、コンピュータからデバイスのネットワーク (ネットワーク名は DeepRacer) に接続できるように正しく設定されていることを確認できます。これを行うには、次の手順に従います。

Note

以下の手順では、MacOS コンピュータを使用していることを前提としています。他のコンピュータシステムの場合は、各オペレーティングシステムのネットワーク設定に関するドキュメントを参照し、一般的なガイドとして以下の手順を使用します。

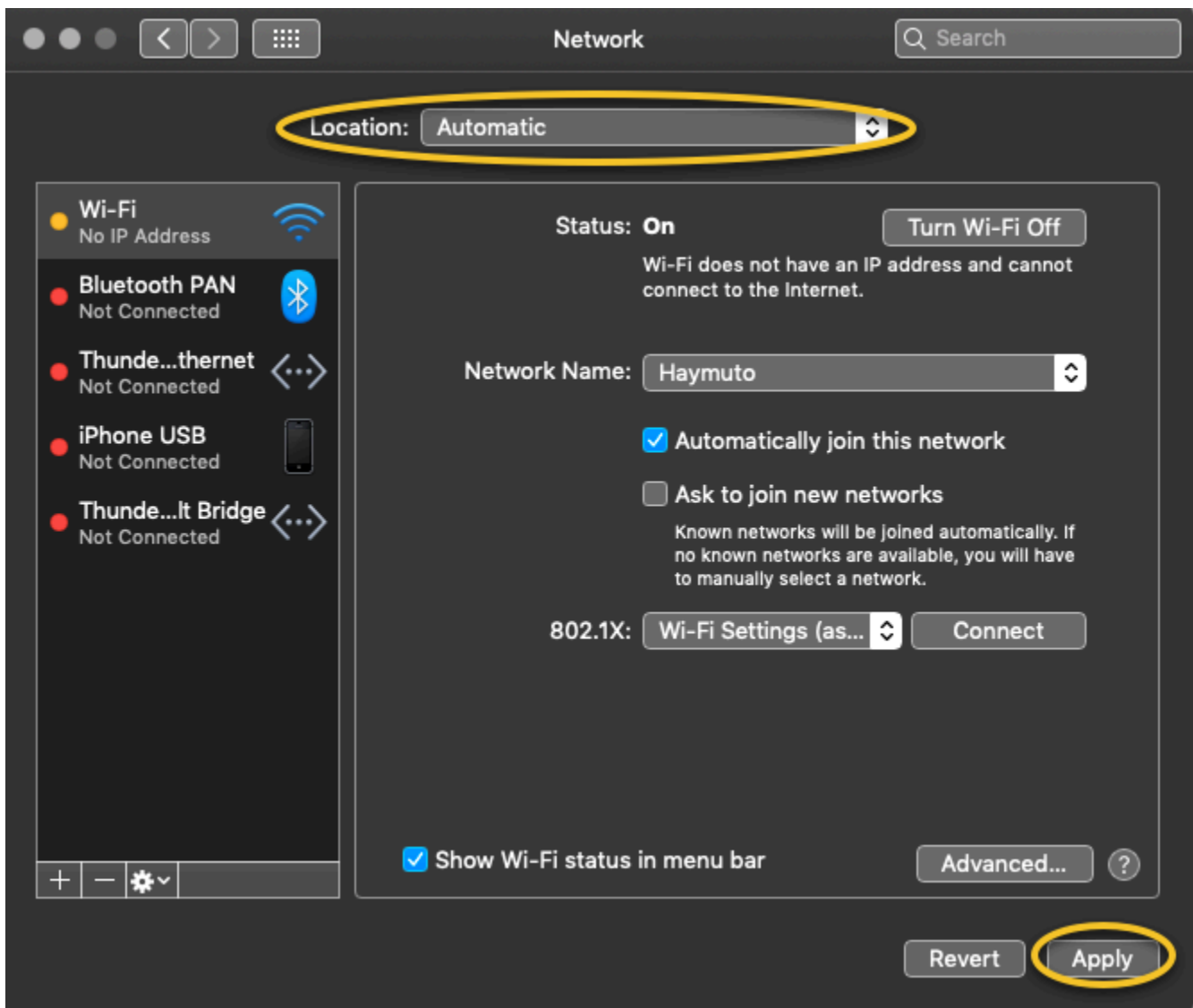
MacOS コンピュータでデバイスの USB-over-ethernet ネットワークをアクティブにするには

1. ネットワークアイコン (ディスプレイの右上) を選択して、[ネットワーク設定] を開きます。

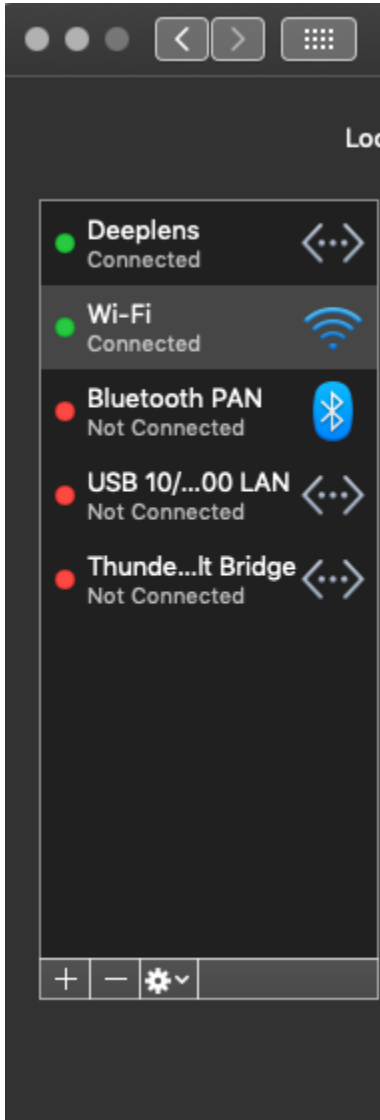


または、Command+space を選択し、「Network」と入力して、[Network System Preferences] を選択します。

2. DeepRacer が [接続済み] と表示されているかどうかを確認します。DeepRacer が表示されているが接続されていない場合は、マイクロ USB/USB ケーブルが車両とコンピュータの間にしっかりと接続されていることを確認します。
3. USB ケーブルが差し込まれているときに DeepRacer ネットワークが表示されていないか、リストに表示されているのに接続されていない場合、[ロケーション] 設定から [自動] を選択し、[適用] を選択します。



4. AWS DeepRacer ネットワークが、[接続済み] として表示していることを確認します。



5. コンピュータが DeepRacer ネットワークに接続されたら、ブラウザの `https://deepracer.aws` ページを更新し、[入門ガイド] の「Wi-Fi への接続」の残りの手順に進みます。
6. DeepRacer ネットワークが接続されていない場合は、コンピュータを AWS DeepRacer 車両から切断し、あらためて接続します。DeepRacer ネットワークが [接続済み] になったら、[入門ガイド] の手順に進みます。
7. デバイスの DeepRacer ネットワークがそれでも接続されない場合は、コンピュータと AWS DeepRacer 車両を再起動し、必要に応じてこの手順のステップ 1 から繰り返します。

上記の対策で問題が解決しない場合は、デバイス証明書が破損している可能性があります。以下のステップに従って、AWS DeepRacer 車両用の新しい証明書を生成し、破損したファイルを修復します。

AWS DeepRacer 車両で新しい証明書を生成するには

1. マイクロ USB/USB ケーブルを取り外し、コンピュータと AWS DeepRacer 車両との USB 接続を終了します。
2. AWS DeepRacer 車両を(HDMI-HDMI ケーブルを使用して) モニターに、また USB キーボードとマウスに接続します。
3. AWS DeepRacer オペレーティングシステムにログインします。これがデバイスのオペレーティングシステムへの最初のログインである場合は、求められたらパスワードとして `deepracer` を使用します。必要に応じてパスワードの変更に進み、更新されたパスワードを以降のログインに使用します。
4. ターミナルウィンドウを開いて、次の Shell コマンドを入力します。ターミナルウィンドウを開くには、デスクトップで、アプリケーション->システムツールから [ターミナル] ショートカットを選択します。または、ファイルブラウザを使用して `/usr/bin` フォルダに移動し、`gnome-terminal` を選択して開きます。

```
sudo /opt/aws/deepracer/nginx/nginx_install_certs.sh && sudo reboot
```

プロンプトが表示されたら、前のステップで使用または更新したパスワードを入力します。

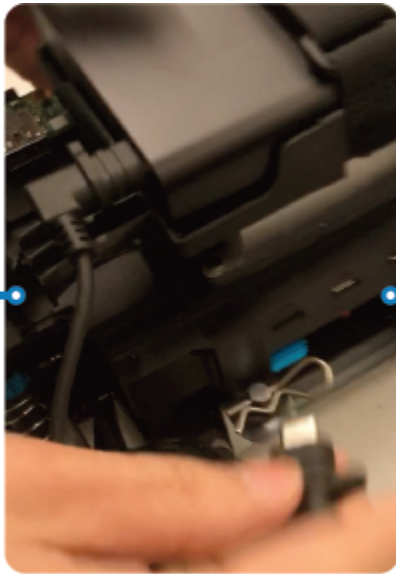
上記のコマンドにより、新しい証明書がインストールされ、デバイスが再起動されます。また、デバイスコンソールのパスワードは、AWS DeepRacer 車両の下部に表示されているデフォルト値に戻されます。

5. モニター、キーボード、マウスを車両から切り離し、マイクロ USB/USB ケーブルでコンピュータに再接続してください。
6. [このトピックの 2 番目の手順](#)に従い、デバイスコンソールを再度開く前に、コンピュータが実際にデバイスネットワーク (`https://deepracer.aws`) に接続されていることを確認し、その後、[入門ガイド] 内の [Wi-Fi 接続] の手順へと進んで下さい。

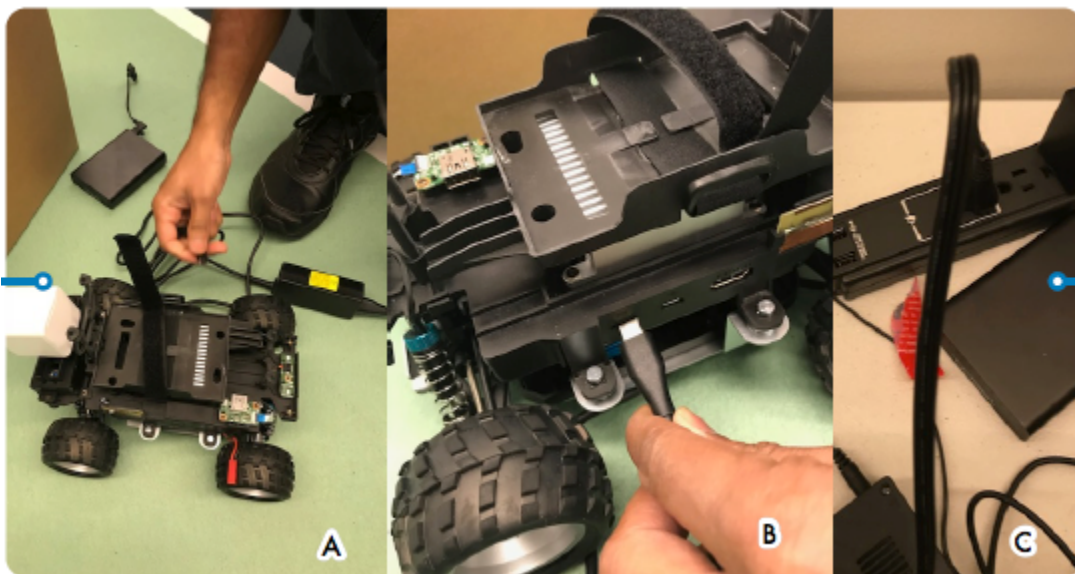
AWS DeepRacer コンピューティングモジュールの電源を、バッテリーからコンセントに切り替える方法

最初に AWS DeepRacer を設定する際にコンピューティングモジュールのバッテリーレベルが低い場合は、以下のステップで、コンピュータの電源をバッテリーからコンセントに切り替えます。

1. USB-C ケーブルを車両のコンピュータ電源ポートから抜きます。



2. AC 電源コードと USB-C ケーブルをコンピュータモジュールの電源アダプター (A) に接続します。電源コードをコンセント (C) に差し込み、USB-C ケーブルを車両のコンピュータモジュールの電源ポート (B) に差し込みます。



USB フラッシュドライブを使用して AWS DeepRacer を Wi-Fi ネットワークに接続する方法

USB フラッシュドライブを使用して、AWS DeepRacer 車両を自宅またはオフィスの Wi-Fi ネットワークに接続するには、以下の手順に従って下さい。

- USB フラッシュドライブ
- 接続したい Wi-Fi ネットワークの名前 (SSID) とパスワード

Note

AWS DeepRacer では、ユーザーのサインインに有効な [キャプチャ](#) 認証を必要とする Wi-Fi ネットワークはサポートされていません。


USB フラッシュドライブを用いて AWS DeepRacer を Wi-Fi ネットワークに接続する方法

1. USB フラッシュドライブをコンピュータに接続します。
2. コンピュータでウェブブラウザを開き、<https://aws.amazon.com/deepracer/usbwifi> へ進みます。このリンクを選択すると、GitHub でホストされている `wifi-creds.txt` という名前のテキストファイルが開きます。

40 lines (39 sloc) | 3.25 KB


```
1 #####
2 #                               Connect the AWS DeepRacer vehicle to Wi-Fi #
3 # File name: wifi-creds.txt #
4 # #
5 # To connect the AWS DeepRacer vehicle to Wi-Fi, type your Wi-Fi name (SSID) and #
6 # password in the appropriate field at the end of this file. Both values are case #
7 # sensitive. #
8 # #
9 # For example: #
10 #     ssid: 'Your-WiFi 100' #
11 #     password: 'Passwd1234' #
12 # #
```

3. `wifi-creds.txt` を自分の USB フラッシュドライブに保存します。使用するウェブブラウザによっては、テキストファイルがコンピュータにダウンロードされ、デフォルトのコードエディタで自動的に開く場合があります。`wifi-creds.txt` が自動的にダウンロードされない場合は、コンテキスト (右クリック) メニューを開き、[名前を付けて保存] を選択してテキストファイルを USB フラッシュドライブに保存します。

 Warning

ファイル名は変更しないでください。

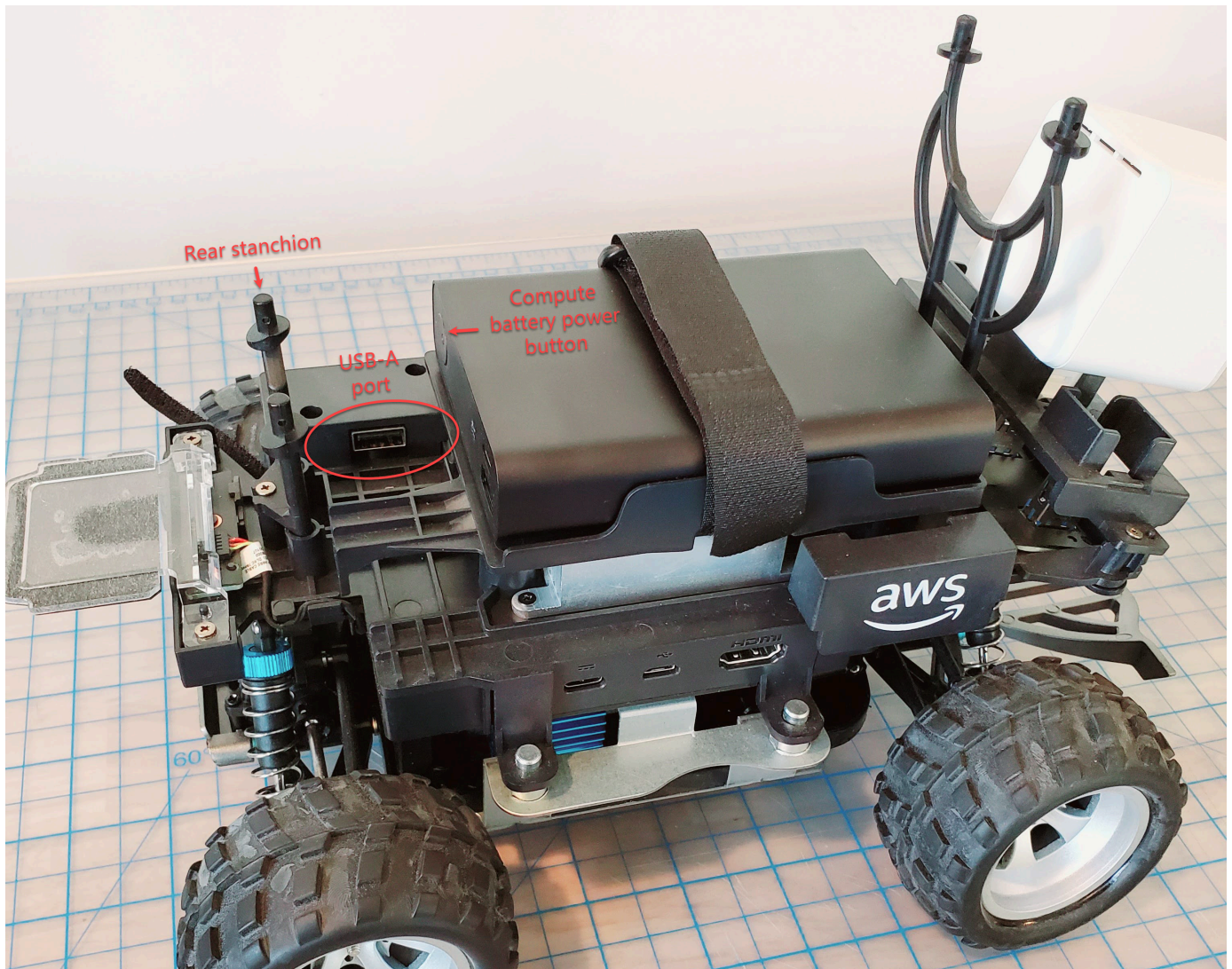
4. `wifi-creds.txt` がまだ開いていない場合は、プレーンテキストモード中のコードエディタで開きます。一部のテキストエディタでは、ファイルの種類が指定されていない場合、プレーンテキスト (`.txt`) ではなくリッチテキスト (`.rtf`) がデフォルトとして表示されるため、ファイルの編集に問題がある場合は、設定を確認します。Windows を使用している場合は、無料でダウンロードできる Sublime Text アプリケーションを使用してファイルを開くこともできます。あるいは Mac を使用している場合は、TextEdit アプリケーションを試して下さい。これはほとんどの Mac デバイスにプリインストールされており、デフォルトではプレーンテキストに設定されています。
5. ファイルの下部にある一重引用符の間に、使用する Wi-Fi ネットワークの名前 (SSID) とパスワードを入力します。SSID は "Service Set Identifier" の略です。Wi-Fi ネットワークの名前を表す専門用語です。

 Note

ネットワーク名 (SSID) またはパスワードに、(My Wi-Fi 100 など) のようにスペースが含まれている場合、スペースを含めて名前を一重引用符 (") の間にそのまま入力します。スペースがない場合、一重引用符の使用はオプションです。例えば、Wi-Fi パスワード `Passwd1234` はスペースを含まないため、一重引用符を使用すると機能しますが、必須ではありません。SSID とパスワードはどちらも大文字と小文字が区別されません。

```
29 # If you have validated the Wi-Fi credentials but the Wi-Fi LED doesn't #
30 # turn solid blue, try restarting the vehicle by pressing the reset button. #
31 # When the power LED turns blue, plug the USB drive in again. #
32 # #
33 # To finish setting up, follow the instructions on https://docs.aws.amazon.com/ #
34 # deepracer/latest/developerguide/deepracer-troubleshooting-wifi-connection-first #
35 # -time. #
36 #####
37
38 # Enter your Wi-Fi name (SSID) and password:
39 ssid: ''
40 password: ''
```

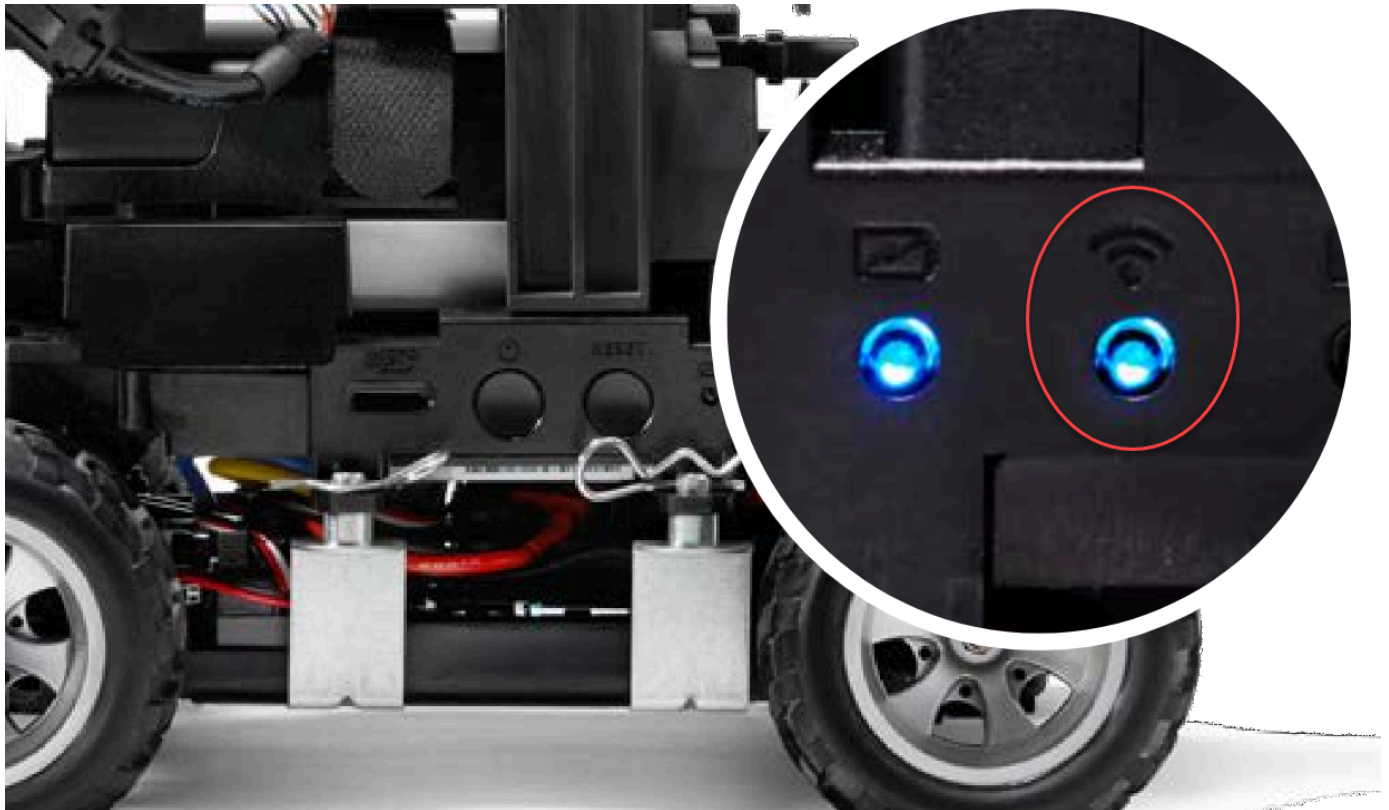
6. ファイルを USB フラッシュドライブに保存します。
7. コンピュータから USB ドライブを取り出し、AWS DeepRacer 車両の後部のコンピュータバッテリーの電源ボタンと後部スタンションとの間にある USB-A ポートに差し込みます。



8. AWS DeepRacer の電源がオンになっていることを確認して下さい。
9. 車両の Wi-Fi LED を確認して下さい。点滅の後、白から青に変われば、車両は Wi-Fi ネットワークに接続されています。USB ドライブを取り外し、ステップ 11 に進んで下さい。

Note

車両を Wi-Fi ネットワークに接続する前に USB ドライブが車両に接続されていた場合は、利用可能な Wi-Fi ネットワークのリストがフラッシュドライブの `wifi-creds.txt` ファイルに自動保存されます。シャープ記号を削除して、接続したいもののコメントを外して下さい。



10. 点滅の後に車両の Wi-Fi LED が赤に変わる場合、USB ドライブを車両から取り外し、コンピュータに接続します。テキストファイルに入力した Wi-Fi 名とパスワードの中に、入力ミス、スペースエラー、文の大文字小文字の誤り、一重引用符の欠落や誤用がないかを確認します。間違いを修正し、ファイルを再保存し、ステップ 7 ~ 9 を繰り返します。
11. 車両の Wi-Fi LED が青に変わったら、USB ドライブを車両から取り外し、コンピュータに接続します。
12. `wifi-creds.txt` ファイルを開きます。テキストファイルの下部で車両の IP アドレスを見つけてコピーします。
13. コンピュータが車両と同じネットワーク中にあることを確認してから、ウェブブラウザに IP アドレスをペーストします。

Note

MacOS Catalina を使用している場合は、Firefox ウェブブラウザを使用します。Chrome はサポートされていません。



14. 接続がプライベートまたは安全でないというメッセージが表示された場合、セキュリティ警告をアクセプトしホストページに進みます。

これで AWS DeepRacer が Wi-Fi に接続されました。

AWS DeepRacer ドライブモジュールのバッテリーを充電する方法

AWS DeepRacer ドライブモジュールバッテリーには、白と赤の 2 つの異なる色の JST コネクター付きの 2 組のケーブルがあります。黒、赤、白のケーブルの端にある白の 3 ピンコネクターは、車両モジュールのバッテリーをバッテリーチャージャーに接続してあります。黒と赤のケーブルの端にある赤色の 2 ピンコネクターで、バッテリーを車両ドライブトレインに接続します。

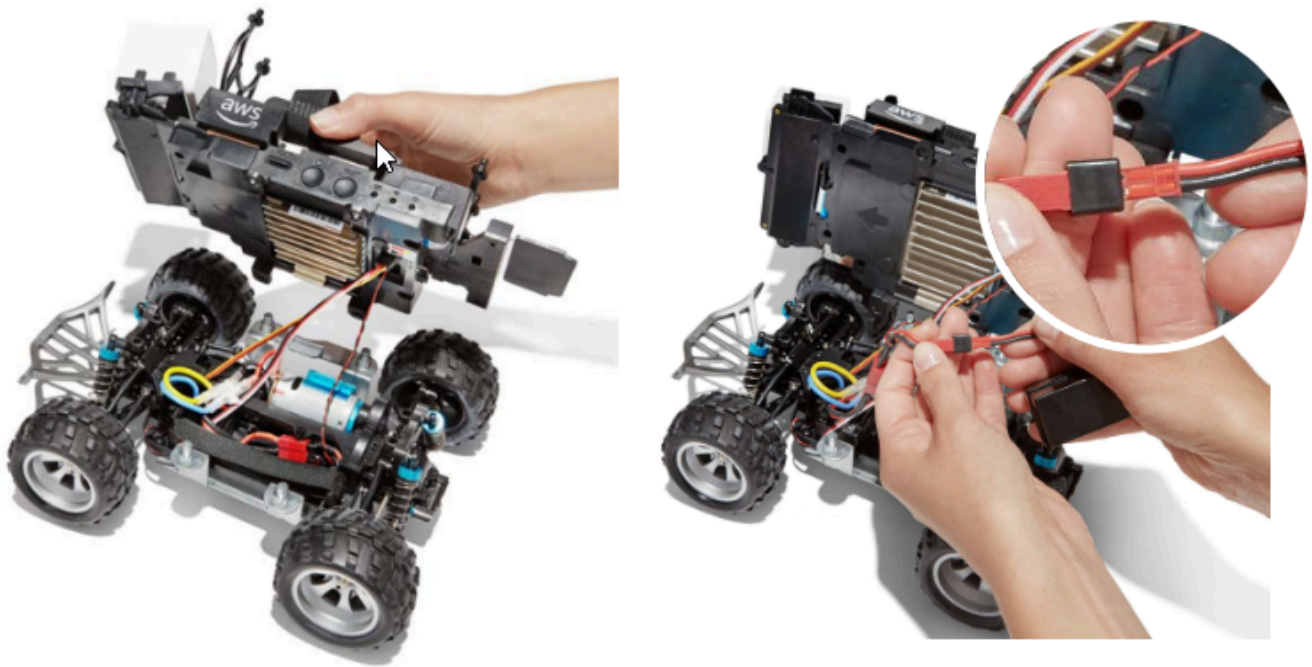


White 3-pin JST Connector
Connects battery to battery charger

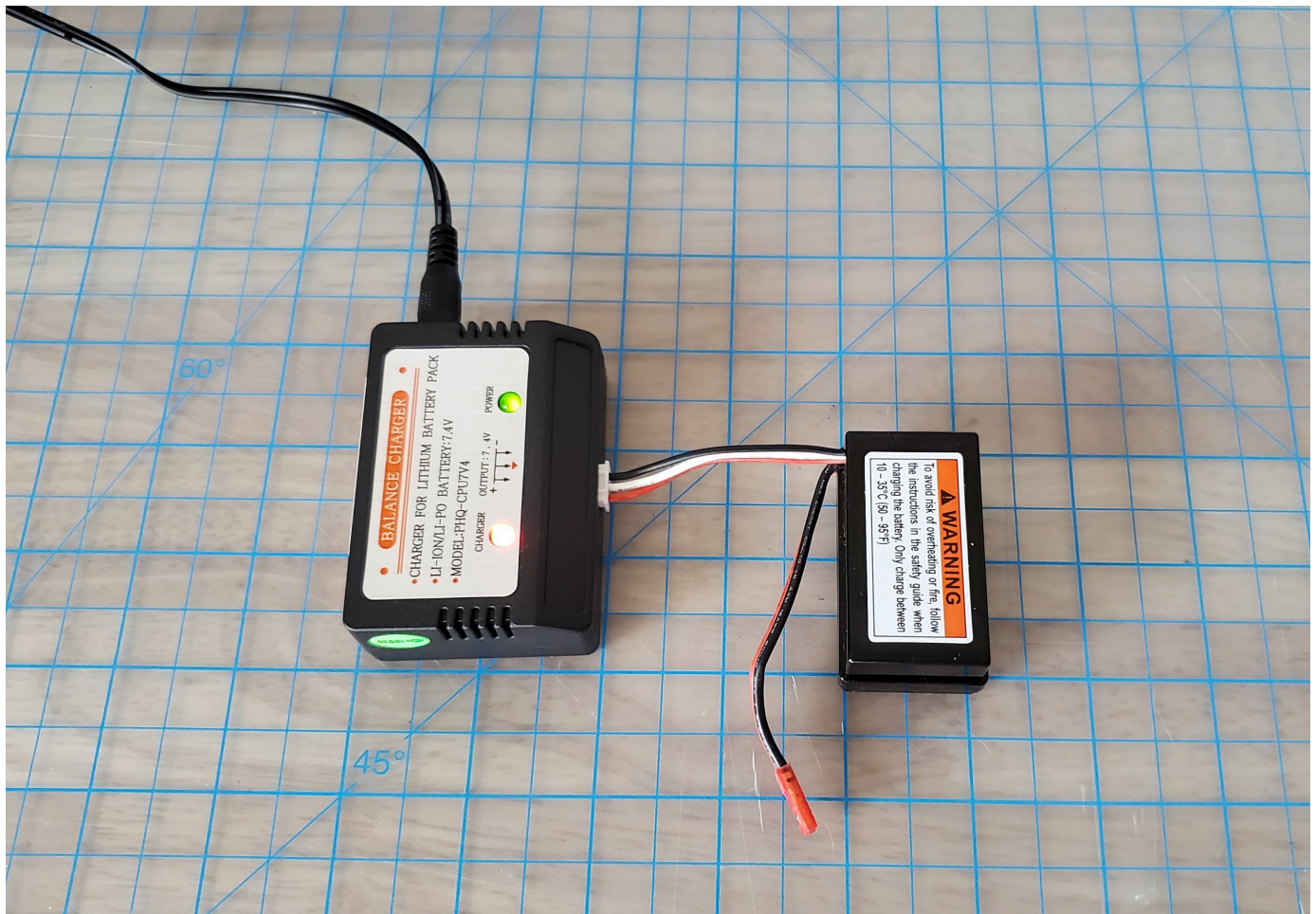
Red 2-pin JST connector
Connects battery to vehicle drive train

以下のステップに従って、AWS DeepRacer ドライブモジュールのバッテリーを充電して下さい。

1. 車両に接続されている場合、ドライブモジュールのバッテリーにアクセスするには、コンピュータモジュールを持ち上げ、ドライブトレインに接続しているワイヤーを緩めないように注意して行ってください。



2. オプションで、ドライブモジュールのバッテリーを車両から取り外すには、赤の2ピンバッテリーコネクタを黒と赤のドライブトレインコネクタから外し、ベルクロストラップを取り外します。
3. バッテリーの白の3ピンコネクタを充電器ポートに接続して、バッテリーをバッテリーチャージャーに取り付けます。



赤色ライト + 緑色のライト = 完全に充電されていません

4. 充電器の電源コードをコンセントに差し込みます。緑色のライトだけが点灯する場合、バッテリーは完全に充電されています。
5. 充電された車両バッテリーの白の3ピンコネクタを充電アダプターから外します。追加事項として、バッテリーを取り外して充電した場合は、赤い2ピンコネクタを車両ドライブトレインコネクタに再接続したことを確認し、ベルクロストラップでバッテリーを車両に固定します。
6. スイッチを「オン」の位置に押して、車両ドライブトレインをオンにします。インジケータ音(2つの短いビープ音)を聞いて、車両が充電されていることを確認します。ビープ音が2回鳴らない場合は、[\[車両のバッテリーのロックを解除する\]](#)を試してください。

これで AWS DeepRacer ドライブモジュールのバッテリーを使用する準備ができました。

AWS DeepRacer コンピューティングモジュールのバッテリーを充電する方法

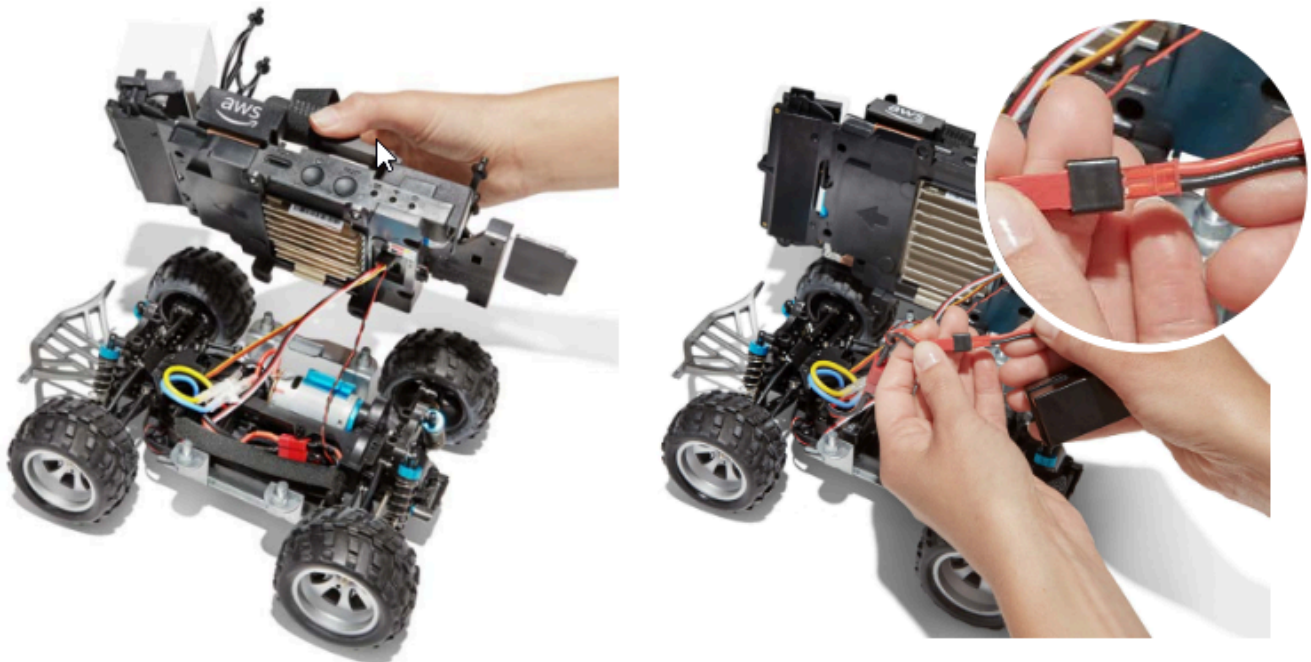
AWS DeepRacer コンピューティングモジュールのバッテリーを充電するには以下のステップに従って下さい。

1. 必要に応じて、コンピューティングモジュールのバッテリーを車両から取り外して下さい。
2. コンピュータの充電器をコンピューティングモジュールのバッテリーに接続します。
3. コンピュータの充電器の電源コードをコンセントに差し込みます。

バッテリーは充電されていますが、AWS DeepRacer 車両が動きません。

AWS DeepRacer コンソールが設定され、コンピューティングバッテリーが充電され、Wi-Fi が接続されているにもかかわらず車両が動かない場合は、次の手順に従ってください。

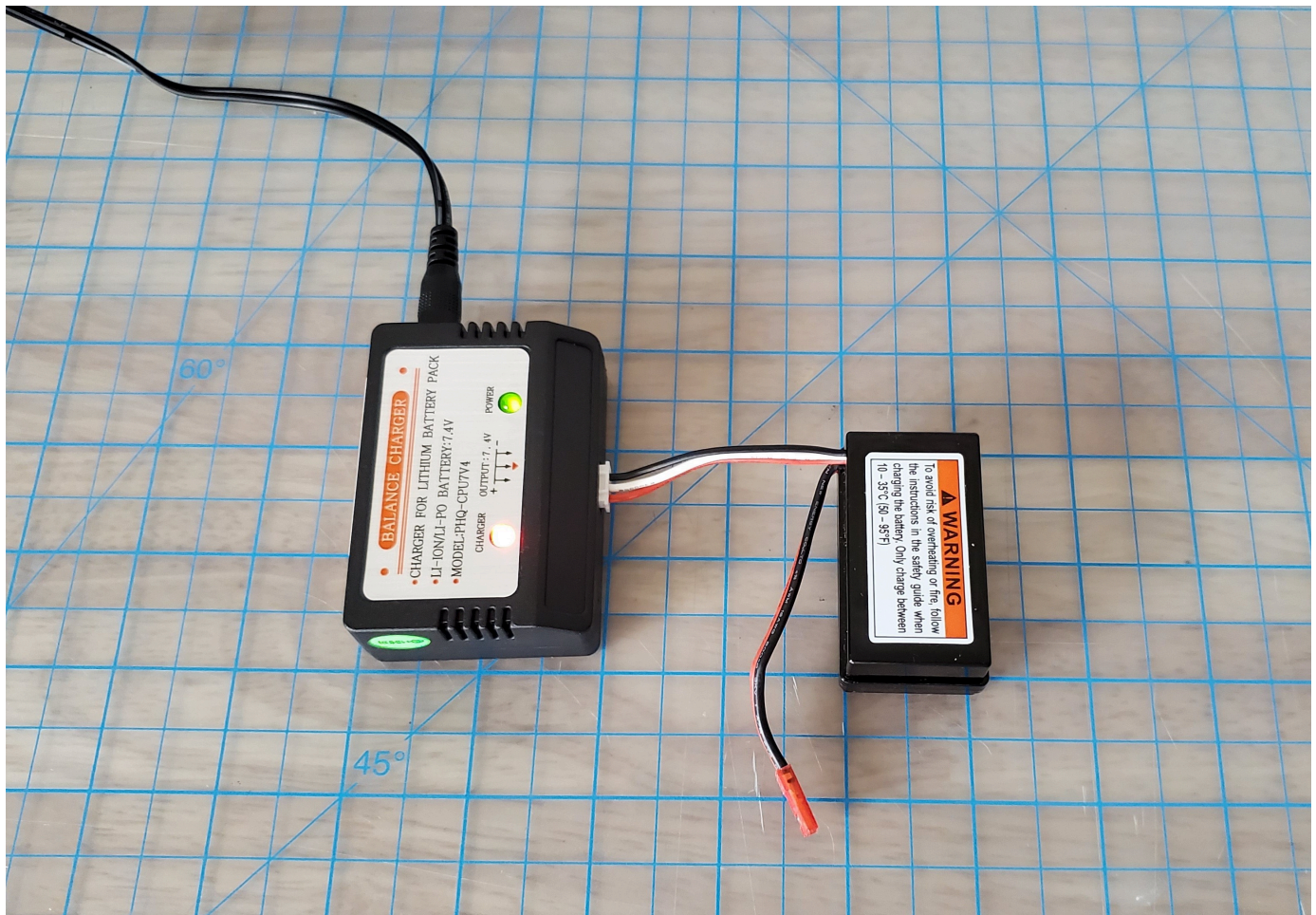
1. コンピューティングモジュールを持ち上げ、ドライブトレインに接続しているワイヤーを緩めないように注意してください。車両の下のバッテリー、赤色の2ピンコネクタが黒と赤のドライブトレインコネクタに正しく接続されていることを確認します。



2. スイッチを「オン」の位置に押し、車両ドライブトレインをオンにします。インジケータ音 (2 つの短いブープ音) を聞いて、車両が充電されていることを確認します。車両の電源が正常に入ったら、ステップ 4 に進みます。



3. 車両のバッテリーを入れたときにブープ音が 2 回鳴らない場合は、バッテリーが完全に充電されていることを確認します。車両のバッテリーの白いコネクタケーブルを充電アダプターに差し込みます。充電アダプターは、赤と緑の LED インジケータライトでコンピューティングモジュールのアダプターと区別できます。アダプターを充電ケーブルに接続し、コンセントに差し込みます。車両用バッテリー充電アダプターの赤と緑の両方が点灯しているときは、バッテリーの充電が必要であることを示します。



赤色ライト + 緑色のライト = 完全に充電 されていません

緑色のライトだけが点灯する場合は、バッテリーは完全に充電されており、すぐに使用できます。充電アダプターから車のバッテリーの白いコネクタを外し、赤いコネクタを車両に再接続して下さい。バッテリーを取り外して充電した場合 (オプション) は、ベルクロストラップでドライブトレインに再度固定します。スイッチを「オン」の位置に押し、車両ドライブトレインをオンにします。それでもビープ音が 2 回鳴らない場合は、[\[車両のバッテリーのロックを解除\]](#) を試してください。

4. 車両を [Wi-Fi](#) に接続し、ブラウザで AWS DeepRacer コンソールを開きます。タッチジョイスティックで車両を手動で運転し、動くことを確認します。

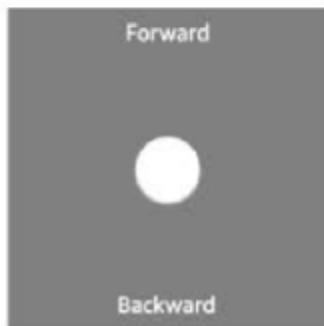
Controls

- Autonomous driving
 Manual driving

Maximum speed



Click or touch to drive



注意: 車両のバッテリーを最も長持ちさせるには、AWS DeepRacer を使用していないときに、車両のドライブトレインのスイッチを切るか、バッテリーを切断するようにします。

それでも車両が動かない場合は、AWSDeepRacer-Help@amazon.com までご連絡ください。

AWS DeepRacer 車両バッテリーのロックアウトのトラブルシューティング

⚠ Important

このバッテリーは DeepRacer カー専用です。このバッテリーは、火災や爆発などの安全上の懸念を回避するために適切に取り扱う必要があります。[[AWS DeepRacer デバイスの安全ガイド](#)]に記載されているすべての指示に従い、すべての警告に従って下さい。

AWS DeepRacer デバイスの利用規約、保証、注意事項

- [AWS DeepRacer デバイスの利用規約](#)
- [AWS DeepRacer デバイスの 1 年間の限定保証](#)

- [AWS DeepRacer デバイスの安全ガイド](#)

トピック

- [AWS DeepRacer 車両バッテリーのロックアウトを防ぐ方法](#)
- [ロックアウト後に AWS DeepRacer 車両のバッテリーをロック解除する方法](#)

AWS DeepRacer 車両バッテリーのロックアウトを防ぐ方法

AWS DeepRacer 車両バッテリーのロックアウトを防ぐ方法について学ぶ。

バッテリーの正常な状態を保つため、AWS DeepRacer 車両バッテリーはロックアウト状態になります。この場合、バッテリー残量があっても、車両には電力が供給されません。車両バッテリーがロックアウト状態になるのを防ぐには、次の操作を行います。

- AWS DeepRacer を使い終わったら、車両の電源を切ってバッテリー充電を保持します。
- デバイスのコンソールに車両のバッテリー残量が低いという警告が表示されたら、できるだけ早く充電します。
- AWS DeepRacer をしばらく使用しない場合は、バッテリーを車両から取り外して完全に充電します。車両バッテリーを保護し、ロックアウトを防ぐために、少なくとも年に 1 回は充電することをお勧めします。

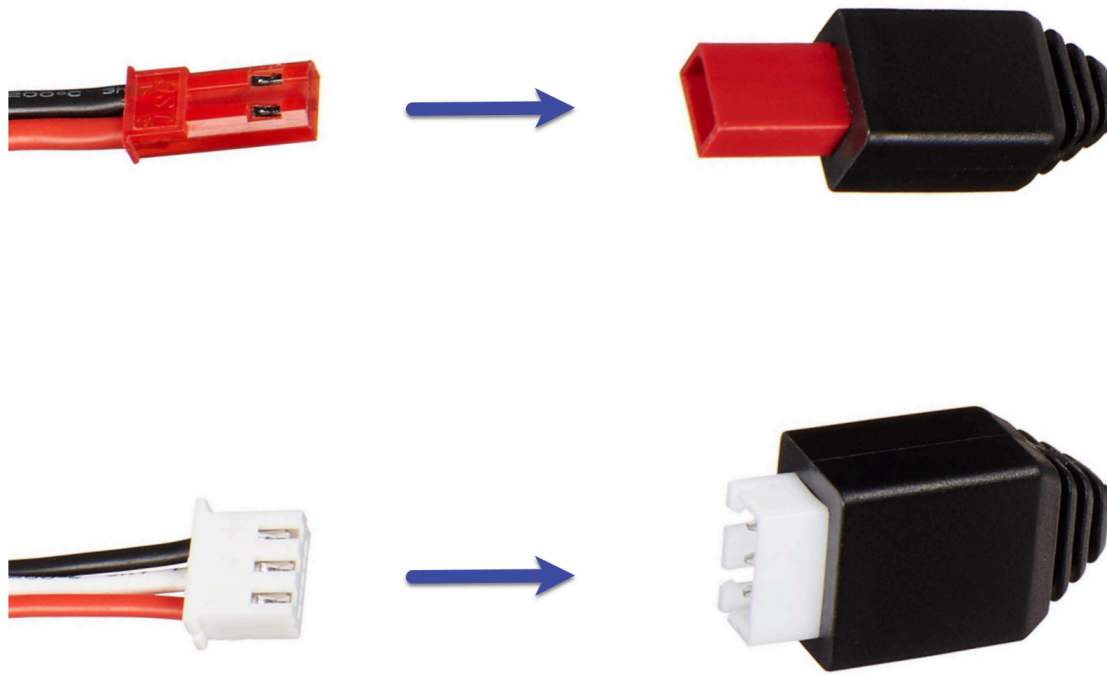
Note

すべてのリチウムポリマー (LiPo) 電池は、使用していないときでも、時間の経過とともに徐々に放電します。

ロックアウト後に AWS DeepRacer 車両のバッテリーをロック解除する方法

ロックアウト後に AWS DeepRacer バッテリーをロック解除するには、[ロック解除ケーブル](#)を使用してください。

1. 赤から赤、白から白へ、同じ色のバッテリーコネクタをケーブルコネクタに挿入します。





2. バッテリーをケーブルから外します。



3. AWS DeepRacer 車両のバッテリーはすぐに使用可能になります。赤い 2 ピンコネクタを車両ドライブトレインコネクタに再接続し、ベルクロストラップでバッテリーを車両に固定します。
4. スイッチを「オン」の位置に押し、車両ドライブトレインをオンにします。インジケータ音 (2 つの短いビープ音) を聞いて、バッテリーが正常にロック解除されたことを確認します。

LiDAR センサーの取り付け時に Dell バッテリーコネクタケーブルを巻きつける方法

非常に長い Dell の USB-C to L 字型 USB-C コネクタケーブルを使用して AWS DeepRacer 車両に接続された LiDAR センサーに Evo シェルを取り付けるには、特別なケーブル装着技術が必要です。

このプロセスのビデオを見るには、YouTube で [\[AWS DeepRacer: LiDAR センサーのインストールと、Dell のコンピューティングバッテリーコネクタケーブルの装着\]](#) を参照してください。ビデオ

は、AWS DeepRacer 車両に LiDAR センサーを取り付けることから始まります。Dell のバッテリー装着テクニックは 00:01:27 秒から始まります。



Note

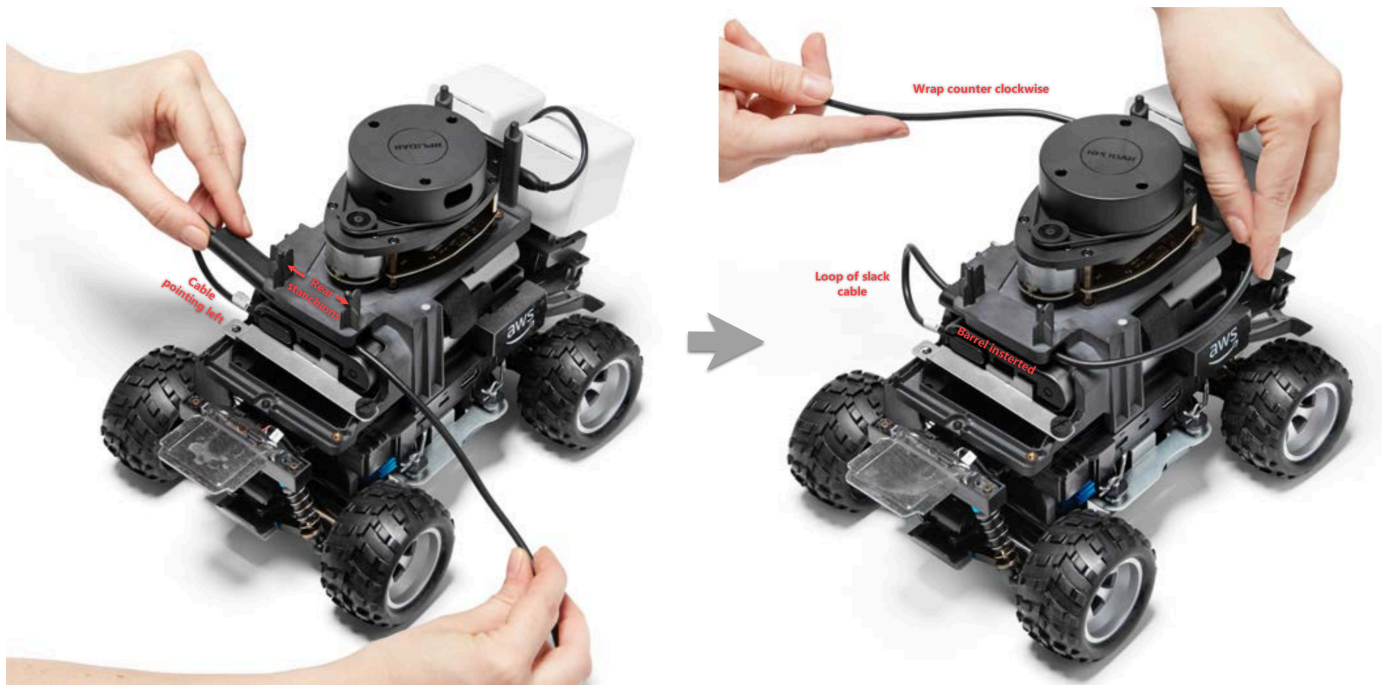
Dell のコンピューティングバッテリーコネクタケーブルには、フェライトコア、L 字型 USB-C、および標準型 USB-C が付属しています。

Evo シェルに対応するために LiDAR センサーに Dell のバッテリーケーブルを巻き付けるには

1. AWS DeepRacer 車両を後ろに向け、コンピューティングバッテリーコネクタケーブルの L 字型端子をコンピューティングバッテリー USB-C ポートに差し込み、コネクタケーブルを左に向けます。



2. 車両を少し左に回して、リア支柱のすぐ下にある LiDAR ホルダーとコンピューティングバッテリーの間のスペースへの開口部を見つけ、ケーブルを通します。フェライトコアがこのスペースに挿入されるまで、ケーブルを引っ張ります。USB-C ポートの左側に緩んだケーブルの輪ができるはずですが。



3. AWS DeepRacer 車両の後ろに向けて LiDAR センサーの土台部分にケーブルを反時計回りに巻き付け、ケーブルクリップを使用してケーブルを固定させ、ぴったりとフィットさせます。
4. 車両を少し右に回し、ケーブルの標準型の USB-C 端子を USB-C ポートに差し込みます。



5. AWS DeepRacer 車両に Evo シェルを置き、ピンでフィットするように締め付けます。シェルが正しくフィットすると、LiDAR センサーがシェルの切り抜きから完全に見えるようになり、支柱の上部にあるピンホールにアクセスできるようになります。シェルを取り外し、必要に応じてケーブルを調整してください。



LiDAR センサーが接続されました。車両の電源を入れ、ドライブ、実験する準備が整いました。

車両の Wi-Fi 接続を維持する方法

次のトラブルシューティングガイドでは、車両の接続を維持するためのヒントを紹介します。

車両の Wi-Fi の LED インジケータが青色に点滅し、その後 2 秒間赤色に変わって、最後にオフになった場合の Wi-Fi 接続のトラブルシューティング方法

以下を確認して、有効な Wi-Fi 接続設定があることを確認してください。

- USB ドライブには、wifi-creds.txt ファイルを 1 つのみ持つ、ディスクパーティションが 1 つのみあることを確認します。複数の wifi-creds.txt ファイルが見つかった場合は、すべてが見つかった順序で処理され、予期しない動作を引き起こす可能性があります。
- Wi-Fi ネットワークの SSID とパスワードが wifi-creds.txt ファイルで正しく指定されていることを確認します。このファイルの例は次のようになります。

```
#####
#                               AWS DeepRacer                               #
# File name: wifi-creds.txt                                             #
#                                                                           #
# ...                                                                     #
#####

# Provide your SSID and password below
ssid: ' MyHomeWi-Fi'
password: myWiFiPassword
```

- wifi-creds.txt ファイル内の ssid そして password の両方のフィールド名が小文字であることを確認します。
- 各フィールド名と値が 1 つのコロン (:) で区切られていることを確認します。例えば、ssid : ' MyHomeWi-Fi'
- スペースを含むフィールド値が一對の一重引用符で囲まれていることを確認します。Mac では、TextEdit または他のテキストエディタでは、'...' 形式の一重引用符が表示されますが、'...' は表示されません。フィールド値にスペースが含まれていない場合は、値を一重引用符なしで指定できます。

車両の Wi-Fi または電源 LED インジケータが青く点滅しているのはどういう意味ですか？

USB ドライブに wifi-creds.txt ファイルが含まれている場合、車両がファイルで指定された Wi-Fi ネットワークに接続しようとしている間は、Wi-Fi LED インジケータが青く点滅します。

USB ドライブに models ディレクトリがある場合、車両がディレクトリ内のモデルファイルをロードしようとしている間、電源 LED が青く点滅します。

USB ドライブに wifi-creds.txt と models ディレクトリの両方がある場合、Wi-Fi に接続してモデルをロードしようとする試みから始めて、車両は 2 つを順次処理します。

Wi-Fi 接続の試行が失敗した場合、Wi-Fi LED も 2 秒間赤色に点灯することがあります。

ホスト名を使用して車両のデバイスコンソールに接続するにはどうすればよいですか？

ホスト名を使用して車両のデバイスコンソールに接続する際、必ずブラウザで「`https://hostname.local`」と入力してください。ここでの *hostname* の値 (AMSS-1234 フォーマット) は AWS DeepRacer 車両の下部に印刷されています。

IP アドレスを使用して車両のデバイスコンソールに接続する方法

IP アドレスを使用して、`device-status.txt` ファイル (USB ドライブ内) にあるようにドライブコンソールに接続するには、次の条件が満たされていることを確認します。

- ノートパソコンまたはモバイルデバイスが AWS DeepRacer 車両と同じネットワークにあることを確認します。
- VPN に接続しているかどうかを確認します。接続している場合は、最初に切断します。
- 別の Wi-Fi ネットワークを試します。たとえば、スマートフォンで個人のホットスポットを有効にします。

AWS DeepRacer デバイスの MAC アドレスを取得する方法

以下の手順に従って AWS DeepRacer デバイスの MAC アドレスを取得します。

1. AWS DeepRacer デバイスが Wi-Fi ネットワークにのみ接続されていることを確認してください。
2. AWS DeepRacer 車両をモニタリングに接続します。HDMI-to-HDMI、HDMI-to-DVI または同様のケーブルが必要になり、ケーブルの一端を車両シャーシの HDMI ポートに挿入し、もう一方の端をモニターのサポートされているディスプレイポートに接続します。
3. コンピューティングモジュールを起動した後で、AWS DeepRacer デバイスのコンピューティングモジュールにある USB ポートを使用して、USB キーボードを AWS DeepRacer デバイスに接続します。
4. [ユーザーネーム] 入力フィールドに `deepracer` と入力します。
5. [パスワード] 入力フィールドに、デバイスの SSH パスワードを入力します。

デバイスに初めてログインする場合は、パスワード入力フィールドに「deepracer」と入力します。必要に応じて、次の手順に進む前にパスワードをリセットします。今後のログインには、新しいパスワードを使用します。セキュリティ上の理由から、新しいパスワードには複雑なまたは強力なパスワードフレーズを選択してください。

6. ログインしたら、ターミナルウィンドウを開きます。

ターミナルアプリケーションには、[検索] ボタンを使用できます。

7. 次のように、Ubuntu シェルコマンドをターミナルウィンドウに入力します。

```
ifconfig | grep HWaddr
```

このコマンドでは、以下のような出力が生成されます。

```
m1an0    Link encap:Ethernet    HWaddr    01:2a:34:b5:c6:de
```

16 進数は、デバイスの MAC アドレスです。

AWS DeepRacer デバイスコンソールのデフォルトパスワードを回復する方法

AWS DeepRacer デバイスコンソールのデフォルトパスワードの回復には、デフォルトのパスワードの取得またはリセットが関係しています。デフォルトのパスワードは、デバイスの底に次のイメージのように表示されます。



次の手順に従い、Ubuntu ターミナルウィンドウを使用して AWS DeepRacer デバイスのウェブサーバーのパスワードを回復します。

1. AWS DeepRacer 車両をモニタリングに接続します。HDMI-to-HDMI、HDMI-to-DVI または同様のケーブルが必要になり、ケーブルの一端を車両シャーシの HDMI ポートに挿入し、もう一方の端をモニターのサポートされているディスプレイポートに接続します。
2. コンピューティングモジュールを起動した後で、デバイスのコンピューティングモジュールにある USB ポートを使用して、USB キーボードを AWS DeepRacer に接続に接続します。
3. [ユーザー名] に `deep racer` と入力します。
4. [パスワード] に、デバイスの SSH パスワードを入力します。

デバイスに初めてログインする場合は、パスワードで `deep racer` と入力します。必要に応じて、次の手順に進む前にパスワードをリセットします。今後のログインには、新しいパスワードを使用します。セキュリティ上の理由から、新しいパスワードには複雑なまたは強力なパスワードフレーズを選択してください。

5. ログインしたら、ターミナルウィンドウを開きます。

[検索] ボタンを使用して、ターミナルウィンドウアプリケーションを見つけることができます。

6. デバイスコンソールのデフォルトのパスワードを取得するには、ターミナルウィンドウに次のコマンドを入力します。

```
$cat /sys/class/dmi/id/chassis_asset_tag
```

このコマンドは、結果としてデフォルトパスワードを出力します。

7. デバイスコンソールのパスワードをデフォルトにリセットするには、ターミナルウィンドウで次の Python スクリプトを実行します。

```
sudo python /opt/aws/deepracer/nginx/reset_default_password.py
```

AWS DeepRacer デバイスを手動で更新するには

AWS DeepRacer サービスの最近の変更により、AWS re:Invent 2018 で配布されたデバイスなど、特定のレガシーデバイスが自動更新できなくなります。そのようなデバイスを手動で更新するには、以下の手順に従います。

AWS DeepRacer デバイスを手動で更新する方法

1. コンピュータにダウンロードし、これを解凍すると [AWS DeepRacer デバイススクリプト](#) を手動で更新します。

このスクリプトの解凍したファイルのデフォルト名は `deepracer-device-manual-update.sh` です。このトピックでは、このデフォルトのスクリプトファイル名を使用します。

2. ダウンロードして解凍したスクリプトファイル (`deepracer-device-manual-update.sh`) をコンピュータから USB ドライブにコピーします。
3. HDMI-HDMI ケーブルを使用してデバイスにモニターを接続します。また、USB キーボード、USB マウスも接続します。
4. デバイスの電源を入れ、デバイスの起動後に OS にサインインします。

デバイスへの初回サインインである場合は、新しい OS パスワードを設定する必要があります。

5. USB ドライブをデバイスに接続し、スクリプトファイルをデバイス上のフォルダ (例: `~/Desktop` など) にコピーします。

6. デバイスのターミナルから以下のコマンドを入力して、スクリプトファイルのフォルダに移動し、スクリプトファイルの実行アクセス許可を追加します。

```
cd ~/Desktop
chmod +x deepracer-device-manual-update.sh
```

7. 以下のシェルコマンドを入力して、スクリプトを実行します。

```
sudo -H ./deepracer-device-manual-update.sh
```

8. デバイスの更新が完了したら、コンピュータまたはモバイルデバイスでウェブブラウザを開き、デバイスの IP アドレス (たとえばホームネットワークでは 192.168.1.11、オフィスネットワークでは 10.56.101.13) に移動します。

デバイスが Wi-Fi ネットワークに接続されていることを確認し、VPN によるトンネリングを介することなく同じネットワークにあるブラウザを使用します。

9. デバイスコンソールで、デバイスコンソールのパスワードを入力してサインインします。更新画面が表示されるまで待ちます。さらに更新を求められたら、その指示に従います。

AWS DeepRacer の一般的な操作問題の診断および解決方法

AWS DeepRacer 車両を使って強化学習を学んでいると、デバイスが機能しなくなることがあります。以下のトラブルシューティングのトピックは、問題の診断と解決に役立ちます。

トピック

- [デバイスコンソールのビデオプレーヤーに車両のカメラからのビデオストリームが表示されないのはなぜですか？](#)
- [なぜ、私の AWS DeepRacer が動かないのでしょうか？](#)
- [デバイスの最新アップデートが表示されないのはなぜですか？最新アップデートを入手する方法を教えてください。](#)
- [AWS DeepRacer 車両が Wi-Fi ネットワークに接続されないのはなぜですか。](#)
- [AWS DeepRacer のデバイスコンソールページの読み込みに時間がかかるのはなぜですか。](#)
- [AWS DeepRacer 車両にデプロイすると、モデルが適切に動作しないのはなぜですか？](#)

デバイスコンソールのビデオプレーヤーに車両のカメラからのビデオストリームが表示されないのはなぜですか？

AWS DeepRacer デバイスコンソールにログインした後で、AWS DeepRacer 車両に搭載されたカメラからのライブビデオストリームが、デバイスコントロールのビデオプレーヤーに表示されません。この問題の原因としては、以下が考えられます。

- カメラの USB ポートへの接続が緩くなっている可能性があります。カメラモジュールを車両から取り外し、USB ポートに再び差し込み、デバイスの電源をオフにしてから再びオンにして再起動します。
- カメラに不具合がある可能性があります。別の AWS DeepRacer 車両の正常に動作しているカメラを使用して、可能なら、これが原因であるかどうかをテストします。

なぜ、私の AWS DeepRacer が 動かないのでしょうか？

AWS DeepRacer 車両の電源を入れても、始動しません。この問題の原因としては、以下が考えられます。

- 車両のパワーバンクがオンになっていないか、パワーバンクが車両に接続されていません。付属の USB-C-to-USB C ケーブルを、電源バンクの USB-C ポートと車両シャーシの USB-C ポートの間に接続してください。LED インジケータが点灯していることを確認します。これは、電源バンクの充電レベルを示します。点灯していない場合は、電源バンクの電源ボタンを押し、車両のシャーシにある電源ボタンを押し、デバイスを起動します。テールライトが点灯していると、デバイスは起動しています。
- 電源バンクがオンになっていて車両が起動しているが、車両が手動モードまたは自律型運転モードのいずれでも動かない場合は、車両シャーシの下にある車両バッテリーが充電されていて、電源が入っていることを確認します。そうでない場合は、車両バッテリーを充電し、バッテリーが完全に充電された後で電源を入れます。
- 車両バッテリーのケーブルコネクタが、デバイス運転モジュールの電源ケーブルコネクタに完全に接続されていません。ケーブルコネクタがしっかりと差し込まれていることを確認します。
- バッテリーケーブルに不具合があります。可能な場合は、このバッテリーを別の正常に動作している車両で使用し、ケーブルの不具合が原因であるかどうかをテストします。
- 車両のバッテリーの電源スイッチがオンになっていません。電源スイッチをオンにし、ビーブ音が 2 回鳴り、続いて長いビーブ音が 1 回聞こえることを確認します。

デバイスの最新アップデートが表示されないのはなぜですか？最新アップデートを入手する方法を教えてください。

AWS DeepRacer 車両のソフトウェアが古いのはなぜですか？

- デバイスでしばらく自動更新が行われていません。[手動更新](#)を行う必要がある可能性があります。
- 車両がインターネットに接続されていません。車両がインターネットにアクセスできる Wi-Fi またはイーサネット ネットワークに接続されていることを確認します。

AWS DeepRacer 車両が Wi-Fi ネットワークに接続されないのはなぜですか。

車両の OS でネットワークステータスを確認すると、私は AWS DeepRacer 車両が Wi-Fi ネットワークに接続されているのかわかりません。これは、次の問題が原因で発生する可能性があります。

- AWS DeepRacer 車両に Wi-Fi が設定されていません。この[セットアップ手順](#)に従って、車両の Wi-Fi ネットワークを設定します。
- 車両がアクティブなネットワーク信号の範囲内にありません。選択した Wi-Fi ネットワークの範囲内で車両を操作してください。
- 車両に事前設定された Wi-Fi ネットワークが使用可能な Wi-Fi ネットワークと一致しません。[セットアップ手順](#)に従って、アクティブな [CAPTCHA](#) を必要としない Wi-Fi ネットワークを設定します。

AWS DeepRacer のデバイスコンソールページの読み込みに時間がかかるのはなぜですか。

AWS DeepRacer 車両のデバイスコンソールを開こうとすると、デバイスコンソールページの読み込みに時間がかかります。

- 車両がダウンしているか、またはオフになっています。テールライトが点灯しているときに、車両の電源が入っていることを確認します。
- 車両の IP アドレスが変更されています。ネットワークの DHCP サーバーによって変更された可能性が大了。車両の新しい IP アドレスを確認するには、この[セットアップ手順](#)に従い、コンピュータと車両との USB-USB ケーブル接続を使用してデバイスコンソールにサインインします。[設定] で新しい IP アドレスを確認します。または、ネットワークに接続されているデバイスのリ

ストを調べて、新しい IP アドレスを確認することもできます。ネットワーク管理者でない場合は、管理者にこの件の調査を依頼してください。

AWS DeepRacer 車両にデプロイすると、モデルが適切に動作しないのはなぜですか？

モデルをトレーニングし、そのアーティファクトを AWS DeepRacer 車両にデプロイした後で、時々車両が正常に動作しないことがあります。何が問題だったのですか？

一般的に、物理的な AWS DeepRacer 車両への転送のためにトレーニングされたモデルを最適化することは、容易ではない学習プロセスです。多くの場合、試行錯誤を繰り返す必要があります。ベストプラクティスの一般的なガイドラインについては、「[実環境に合わせた AWS DeepRacer モデルのトレーニングを最適化する](#)」を参照してください。

以下は、AWS DeepRacer 車両のモデルパフォーマンスに影響する可能性のある一般的な要因です。

- モデルがトレーニングにコンバートしていない。モデルのクローンを作成してトレーニングを継続するか、より長い期間モデルを再トレーニングします。エージェントがシミュレーションのラップを連続して終了する (つまり、トレーニングの最後まで 100% 処理する) ことを確認します。
- モデルが過剰トレーニング (つまり、過適合) されています。トレーニングデータに適合しすぎて、未知の状況に対する汎用性がありません。より柔軟または寛容な [報酬関数](#) でモデルを再トレーニングするか、[アクションスペース](#) の粒度を高めます。トレーニング済みのモデルをさまざまなトラックで評価し、モデルが十分な汎用性を備えていることを確認してください。
- AWS DeepRacer 車両が正しくキャリブレートされていない可能性があります。これが該当するかどうかをテストするには、手動運転に切り替えて、車両が正常に走行するかどうかを確認します。正常に走行しない場合は、[車両をキャリブレート](#) します。
- 要件を満たさないトラックで自律型車両を運転しています。トラックの要件については、「[AWS DeepRacer の物理的なトラックを構築する](#)」を参照してください。
- 物理トラックの近くにある物体が多すぎるため、トラックがシミュレートされた環境と大きく異なります。トラックの周囲を片付けて、物理トラックをシミュレートされたトラックにできるだけ近づけます。
- トラック表面または近くの物体からの反射により、カメラを混乱させるグレアが生じる可能性がある。照明を調整し、滑面コンクリートや他の光沢素材をトラックに使用しないようにします。

AWS DeepRacer 開発者ガイドのドキュメント履歴

以下の表に、AWS DeepRacer の前回のリリース以後に行われたドキュメントの重要な変更を示します。

変更	説明	日付
2023 AWS DeepRacer League の更新	AWS DeepRacer League を参照する複数のトピックを更新しました。2023 AWS DeepRacer League シーズンの詳細については、「 利用規約 」をご覧ください。	2023 年 3 月 1 日
SageMaker AI ノートブックを使用した AWS DeepRacer モデルのトレーニングと評価のトピックが一時的に削除されました	トピック「SageMaker AI ノートブックを使用した AWS DeepRacer モデルのトレーニングと評価」が削除されました。 モデルのトレーニングと評価 。現在、AWS DeepRacer AWS で SageMaker AI ノートブックを使用する手順が更新されています。	2022 年 11 月 1 日
マルチユーザー機能の IAM 管理ポリシーの更新	新しい管理ポリシー <code>AWSDeepRacerAccountAdminAccess</code> と <code>AWSDeepRacerDefaultMultiUserAccess</code> が追加され、マルチユーザーモードを使用して 1 つの AWS DeepRacer アカウントで複数の参加者をスポンサーできるようにします。「 the section called “AWSDeepR	2021 年 10 月 26 日

[acerAccountAdminAccess”](#)」

を参照して下さい。

[マルチユーザー機能の更新](#)

AWS DeepRacer は、1 つの AWS アカウントが複数の参加者をスポンサーしてレースやトレーニングを行うことができるマルチユーザー機能をサポートするようになりました。詳細については、「[マルチユーザーモード](#)」を参照してください。

2021 年 10 月 26 日

[マルチカーレースと障害物回避の更新](#)

AWS DeepRacer は、マルチカーレースと障害物回避を可能にする新しいタイプのステレオカメラと LIDAR をサポートするようになりました。詳細については、「[the section called “レースの種類とセンサーの有効化”](#)」を参照してください。

2019 年 12 月 2 日

[コミュニティレースの更新](#)

AWS DeepRacer では、ユーザーがコミュニティレースと呼ばれる独自のレースイベントを組織できるようになり、プライベートのリーダーボードは招待されたユーザーのみに公開されています。詳細については、「[レースに参加する](#)」を参照してください。

2019 年 12 月 2 日

[一般的な可用性の更新](#)

AWS DeepRacer は、深層学習モデルをトレーニングおよび評価するためのより堅牢な方法の特徴としています。ユーザーインターフェースが更新され、説明されています。独自の物理的なトラックを構築するために、より多くのオプションと正確なデータが利用可能になりました。トラブルシューティング情報が利用できるようになりました。

2019 年 4 月 29 日

[「AWS DeepRacer 開発者ガイド」の初回リリース](#)

AWS DeepRacer コンソールや AWS DeepRacer シミュレーター、そして AWS DeepRacer のスケールモデルの車両を使用した、AWS DeepRacer ユーザーのための強化学習や自律型レース用アプリケーションの解明に有効なドキュメントの初回リリース。

2018 年 11 月 28 日

AWS 用語集

最新の AWS 用語については、AWS の用語集 リファレンスの[AWS 用語集](#)を参照してください。

翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。