

# USER MANUAL

## Generator Reinforcement Learning Cases Package

This is the user manual for "GeneratorReinforcementLearningCases.unitypackage".

### Unity Requirements

In order to use the package, the user must have the following versions installed in Unity:

- ML Agents (v. 2.0.1)
- New Input System (v.1.7.0)

### Python Requirements

These versions are required to use neural networks and connect with Unity's ML Agents. Due to connectivity issues between different libraries, we recommend using the following versions:

- Python (v. 3.9.0)
- NumPy (v. 2.1.0)
- Onnx (v. 1.16.2)
- Pytorch (v. 2.4.0+cpu)
- ML-Agents (v. 0.30.0)

### Installation

Steps to follow:

1. Open the console.
2. Check your Python version. Use the command `$py` or `$python`.
3. Copy the location of the directory where your Unity project is located.
4. Create the Python environment. Use the command `$py -m venv venv`.
5. Activate the environment. Use the command `$venv\Scripts\activate`.
6. Update the package manager (pip). Use `$python -m pip install --upgrade pip`.
7. Install the ML Agents library for Python. Use `$pip install mlagents`.
8. Install protobuf. Use `$pip install protobuf==3.20.3`.
9. Install packaging. Use `$pip install packaging`.
10. If you want to check for missing extensions, type `$mlagents-learn help`.

# Steps to Train an Agent

To train a neural network, simply create an agent in Unity that uses ML Agents classes.

```
using Unity.MLAgents;  
using Unity.MLAgents.Actuators;  
using Unity.MLAgents.Sensors;
```

Then follow these steps:

1. Type this command `$mlagents-learn` and press play in Unity. The neural network will begin training.
2. If you want to run a specific ID, type `$mlagents-learn --run-id=`.
3. If you want to run a specific configuration, type `$mlagents-learn config/moveToGoal.yaml --run-id=RandomTest`.