

CMPE 260/297- Reinforcement Learning

Instructor: Jahan Ghofraniha

Before installation of each software make sure you generate a new environment and activate it.

Software Installation Guide (Pytorch & Open AI Gym):

Making a new environment in Ananconda:

1. Depending on the installation of Conda on your computer and if you can run it from any folder on your system, you may have to run the conda prompt first and then run the rest of the commands in the following steps.
2. Use conda command to make a new environment called torch (or any name you wish to call your pytorch environment). For conda cheat sheet please refer to the following link:

https://docs.conda.io/projects/conda/en/latest/_downloads/1f5ecf5a87b1c1a8aaf5a7ab8a7a0ff7/conda-cheatsheet.pdf

```
conda create -n torch python=3 pip
```

*It is important to include pip at the end of the command to install pip within the new environment otherwise your installation of libraries in the new environment could impact the base environment of Python.

3. Check the list of your environments then activate your Pytorch environment

```
conda env list
```

 (to get the list of your current environments. The new environment should appear in the list) .

```
conda activate torch
```

 (to activate the environment)

4. Pytorch Installation (Mac version)
 - a. Download Pytorch from Anaconda.org (<https://anaconda.org/pytorch/pytorch>)

pytorch / packages / pytorch 1.2.0+cu92

PyTorch is an optimized tensor library for deep learning using GPUs and CPUs.

Conda Files Labels Badges

License: BSD 3-Clause
Home: <http://pytorch.org/>
2050490 total downloads
Last upload: 7 days and 42 minutes ago

Installers

conda install ?

linux-64 v1.2.0+cu92
win-64 v1.2.0
osx-64 v1.2.0

To install this package with conda run:
`conda install -c pytorch pytorch`

There might be a newer version of Pytorch available from Anaconda.org website. You can type the above command in a terminal (on Mac) or in windows command prompt with administrative privileges. Another option is to use Anaconda package and install Pytorch and its relevant libraries through anaconda:

Anaconda Navigator

File Help

ANACONDA NAVIGATOR

Sign in to Anaconda Cloud

Search Environments

base (root)
Anaconda3
Optimization
gym
optimization
pulpOpt
torch

Create Clone Import Remove

Not installed Channels Update index... Pytorch X

Name	Description	Version
<input type="checkbox"/> _pytorch_select		1.2.0
<input checked="" type="checkbox"/> pytorch	Pytorch is an optimized tensor library for deep learning using gpus and cpus.	1.0.1









2 packages available matching "Pytorch" 1 package selected

Apply Clear

The second software you will need to install is Open AI Gym. This package is not native to Windows (it is developed for Linux-based systems), however, there is a workaround that is explained below. For Mac/Linux users you can just use your Mac or Linux package manager to install it.

You can install Gym from Anaconda.org as well. Type Gym in the search bar in anaconda.org. You should get to the following screen:

Filters
 Type: All ▾ Access: All ▾ Platform: All ▾

⬆ Favorites	⬇ Downloads	📦 Package (owner / package)	Platforms
2	5248	 conda-forge / gym <small>0.17.2</small> The OpenAI Gym: A toolkit for developing and comparing your reinforcement learning agents.	linux-64 osx-64 win-64 <small>conda</small>
8	3285	 akode / gym <small>0.10.5</small> The OpenAI Gym: A toolkit for developing and comparing your reinforcement learning agents.	osx-64 <small>conda</small>
0	2122	 conda-forge / gym-box2d <small>0.17.2</small> The OpenAI Gym: A toolkit for developing and comparing your reinforcement learning agents.	linux-64 osx-64 win-64 <small>conda</small>
0	2042	 conda-forge / gym-classic_control <small>0.17.2</small> The OpenAI Gym: A toolkit for developing and comparing your reinforcement learning agents.	linux-64 osx-64 win-64 <small>conda</small>
0	1913	 conda-forge / gym-atarl <small>0.17.2</small> The OpenAI Gym: A toolkit for developing and comparing your reinforcement learning agents.	linux-64 osx-64 <small>conda</small>
0	1023	 HCC / gym <small>0.12.1</small> The OpenAI Gym: A toolkit for developing and comparing your reinforcement learning agents.	linux-64 osx-64 <small>conda</small>
1	793	 powerai / gym <small>0.15.4</small> A toolkit for developing and comparing reinforcement learning algorithms	noarch <small>conda</small>
0	172	 tkharrat / r-gym <small>0.1.0</small> OpenAI Gym is a open-source Python toolkit for developing and comparing reinforcement learning algorithms. This is a wrapper for the OpenAI Gym API, and enables access to an ever-growing variety of environments. For more details on OpenAI Gym, please see here: <https://github.com/openai/gym>. For more details on the OpenAI Gym API specification, please see here: <https://github.com/openai/gym-http-api>.	linux-64 <small>conda</small>

Click on the first link to get the installation page:







The OpenAI Gym: A toolkit for developing and comparing your reinforcement learning agents.

Conda


Files


Labels

Badges


 License: MIT
 Home: <https://github.com/openai/gym>
 Development: <https://github.com/openai/gym>
 Documentation: <https://gym.openai.com/docs/>
 5213 total downloads
 Last upload: 8 days and 11 hours ago

Installers


conda install 

 linux-64

v0.17.2

 win-64

v0.17.2

 osx-64

v0.17.2

To install this package with conda run:

```
conda install -c conda-forge gym
```

Description

The OpenAI Gym: A toolkit for developing and comparing your reinforcement learning agents.

Copy and paste the installation command in a terminal or command prompt to install Gym.

You are all set for windows/Mac installation of Gym.

To test if everything is working, try the sample code below:

```
import gym

env = gym.make('CartPole-v0')
#env = gym.make('Taxi-v2')
env.reset()

for _ in range(1000):
    env.render()
    #    env.step(env.action_space.sample())
    env.reset()

env.close()
```

This should display an inverted pendulum that is being balanced.

