

Setting Up the Programming Environment (Ubuntu)

The Deep Learning course includes 3 programming assignments which you will need to finish to complete the course. We will use `Python` and `PyTorch` for the programming assignments. This instruction will help you set up the programming environment on your laptops. There are two ways. One is installation using `pip`, the other is installation using `conda` (the recommended way).

- **Installation using pip**

If you already have a working installation of Python 3, you should be able to install `NumPy`, `matplotlib`, `scikit-learn`, `SciPy`, `jupyter`, `PyTorch` and other packages using `pip`. (While Python 3.x is installed by default on Linux, `pip` is not installed by default. You can install `pip` on Ubuntu using `sudo apt install python3-pip`)

```
pip3 install numpy
pip3 install matplotlib
pip3 install scipy
pip3 install scikit-learn
pip3 install jupyter
```

For `PyTorch`, follow the instructions on <https://pytorch.org/> to install from pip repository corresponding to your system. CUDA is not necessary in this course.

In the above commands, you can replace `pip3` with `python3 -m pip` to make sure you are installing the packages for the version of python your system is currently using.

PyTorch Build	Stable (1.4)		Preview (Nightly)	
Your OS	Linux	Mac		Windows
Package	Conda	Pip	LibTorch	Source
Language	Python		C++ / Java	
CUDA	9.2	10.1		None
Run this Command:	pip install torch==1.4.0+cpu torchvision==0.5.0+cpu -f https://download.pytorch.org/whl/torch_stable.html			

- **Installation using conda**

However, the recommended way of configuring your system is by using a `conda` environment.

We recommend that you install the latest version of `Anaconda` from <https://www.anaconda.com/> or `Miniconda` from <https://docs.conda.io/en/latest/miniconda.html>. If you install Anaconda, `NumPy`, `matplotlib`, `scikit-learn`, `SciPy`, and `jupyter` will be installed automatically for the `base` environment.

Anaconda 2019.10 for Linux Installer

Python 3.7 version

Download

64-Bit (x86) Installer (506 MB)
64-Bit (Power8 and Power9) Installer (320 MB)

Python 2.7 version

Download

64-Bit (x86) Installer (477 MB)
64-Bit (Power8 and Power9) Installer (295 MB)

Here is the main procedure of installing Miniconda and PyTorch on Ubuntu.

1. Change the path to the location where `Miniconda3-latest-Linux-x86_64.sh` is stored.

```
# luojing @ DESKTOP-8RP4OLK in /mnt/c/Users/luojing [16:57:05]
$ cd /mnt/e/Installer/linux

# luojing @ DESKTOP-8RP4OLK in /mnt/e/Installer/linux [16:58:23]
$ ll
total 8.0G
-rwxrwxrwx 1 luojing luojing 506M Oct 16 00:20 Anaconda3-2019.10-Linux-x86_64.sh
-rwxrwxrwx 1 luojing luojing 462M Oct 16 00:23 Anaconda3-2019.10-Windows-x86_64.exe
-rwxrwxrwx 1 luojing luojing 17K Jan 12 10:37 DownloadZip.html
-rwxrwxrwx 1 luojing luojing 69M Oct 26 03:35 Miniconda3-latest-Linux-x86_64.sh
-rwxrwxrwx 1 luojing luojing 2.4G Sep 20 18:43 manjaro-kde-18.1.0-stable-x86_64.iso
-rwxrwxrwx 1 luojing luojing 2.3G Sep 11 23:11 manjaro-xfce-18.1.0-stable-x86_64.iso
-rwxrwxrwx 1 luojing luojing 90M Sep 17 21:28 otp_win64_22.1.exe
-rwxrwxrwx 1 luojing luojing 1.1M Sep 4 23:42 rufus-3.6.exe
drwxrwxrwx 1 luojing luojing 4.0K Oct 20 12:41 rufus_files
-rwxrwxrwx 1 luojing luojing 2.3G Oct 17 20:54 ubuntu-19.10-desktop-amd64.iso
```

2. Install Miniconda. Use the "space" key to read the license quickly.

```
# luojing @ DESKTOP-8RP4OLK in /mnt/e/Installer/linux [16:58:25]
$ zsh Miniconda3-latest-Linux-x86_64.sh

Welcome to Miniconda3 4.7.12

In order to continue the installation process, please review the license
agreement.
Please, press ENTER to continue
>>>
=====
Miniconda End User License Agreement
=====

Copyright 2015, Anaconda, Inc.

All rights reserved under the 3-clause BSD License:
```

3. Choose the location where Miniconda will be installed.

```

Do you accept the license terms? [yes|no]
[no] >>> yes

Miniconda3 will now be installed into this location:
/home/luojing/miniconda3

- Press ENTER to confirm the location
- Press CTRL-C to abort the installation
- Or specify a different location below

[/home/luojing/miniconda3] >>>
PREFIX=/home/luojing/miniconda3
Unpacking payload ...
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##

environment location: /home/luojing/miniconda3

```

4. Initialize Miniconda3, which will modify the `.bashrc` file. If you use `zsh` or `fish`, you need to modify the corresponding file manually.

```

installation finished.
Do you wish the installer to initialize Miniconda3
by running conda init? [yes|no]
[no] >>> yes
no change      /home/luojing/miniconda3/condabin/conda
no change      /home/luojing/miniconda3/bin/conda
no change      /home/luojing/miniconda3/bin/conda-env
no change      /home/luojing/miniconda3/bin/activate
no change      /home/luojing/miniconda3/bin/deactivate
no change      /home/luojing/miniconda3/etc/profile.d/conda.sh
no change      /home/luojing/miniconda3/etc/fish/conf.d/conda.fish
no change      /home/luojing/miniconda3/shell/condabin/Conda.psm1
no change      /home/luojing/miniconda3/shell/condabin/conda-hook.ps1
no change      /home/luojing/miniconda3/lib/python3.7/site-packages/xontrib/conda.xsh
no change      /home/luojing/miniconda3/etc/profile.d/conda.csh
modified       /home/luojing/.bashrc

==> For changes to take effect, close and re-open your current shell. <==

If you'd prefer that conda's base environment not be activated on startup,
set the auto_activate_base parameter to false:

conda config --set auto_activate_base false

Thank you for installing Miniconda3!

```

5. Then you can create a conda environment for the course using (It is **optional**. You can also use the `base` environment.)

```

conda create -n cs324 python=3.7
# cs324 is the name of the conda environment. It can be modified.

```

```

(base)
# luojing @ DESKTOP-8RP4OLK in ~ [17:23:45]
$ conda create -n cs324 python=3.7
Collecting package metadata (current_repodata.json): done
Solving environment: done

==> WARNING: A newer version of conda exists. <==
  current version: 4.7.12
  latest version: 4.8.2

Please update conda by running

  $ conda update -n base -c defaults conda

## Package Plan ##

  environment location: /home/luojing/miniconda3/envs/cs324

  added / updated specs:
    - python=3.7

```

6. To activate this environment, use

```
conda activate cs324
```

To deactivate this environment, use

```
conda deactivate
```

You can use `conda list` to list the installed packages in the environment.

```

Executing transaction: done
#
# To activate this environment, use
#
#   $ conda activate cs324
#
# To deactivate an active environment, use
#
#   $ conda deactivate
#

(base)
# luojing @ DESKTOP-8RP4OLK in ~ [19:08:46]
$ conda activate cs324
(cs324)
# luojing @ DESKTOP-8RP4OLK in ~ [19:24:00]
$ conda list
# packages in environment at /home/luojing/miniconda3/envs/cs324:
#
# Name                                Version           Build    Channel
_libcxx_mutex                         0.1               main
ca-certificates                      2020.1.1          0
certifi                              2019.11.28        py37_0
ld_impl_linux-64                     2.33.1            h53a641e_7

```

7. Finally, install the required packages:

```
conda activate cs324

# cpu only
conda install pytorch torchvision cpuonly -c pytorch

# gpu cuda 10.1
conda install pytorch torchvision cudatoolkit=10.1 -c pytorch

# when you install PyTorch, numpy will be installed automatically.
# Now you only need to install other packages.
conda install matplotlib
conda install scipy
conda install scikit-learn
conda install jupyter
```

8. Run `jupyter notebook`.

```
conda activate cs324

jupyter notebook
```

Please note that if you want to use GPU, you need to make sure that the GPU driver has been installed correctly and then install `PyTorch` with `cuda`. You can use `nvidia-smi` to check the GPU driver.

```
(base) luojing@luojing-Z390-UD:~$ nvidia-smi
Sun Feb 16 11:32:42 2020
```

+-----+-----+-----+-----+-----+-----+									
NVIDIA-SMI 435.21		Driver Version: 435.21				CUDA Version: 10.1			
+-----+-----+-----+-----+-----+-----+									
GPU	Name	Persistence-M	Bus-Id	Disp.A	Volatile	Uncorr.	ECC		
Fan	Temp	Perf	Pwr:Usage/Cap	Memory-Usage	GPU-Util	Compute	M.		
+-----+-----+-----+-----+-----+-----+									
0	GeForce RTX 208...	Off	00000000:01:00.0	On			N/A		
0%	44C	P8	12W / 250W	362MiB / 11016MiB	0%		Default		
+-----+-----+-----+-----+-----+-----+									

+-----+-----+-----+-----+-----+-----+					
Processes:					GPU Memory
GPU	PID	Type	Process name	Usage	
+-----+-----+-----+-----+-----+-----+					
0	991	G	/usr/lib/xorg/Xorg		24MiB
0	1375	G	/usr/lib/xorg/Xorg		118MiB
0	25648	G	/usr/bin/gnome-shell		163MiB
+-----+-----+-----+-----+-----+-----+					

When you choose the version of `cuda`, you need to check the version of GPU driver.

Table 1. CUDA Toolkit and Compatible Driver Versions

CUDA Toolkit	Linux x86_64 Driver Version	Windows x86_64 Driver Version
CUDA 10.2.89	>= 440.33	>= 441.22
CUDA 10.1 (10.1.105 general release, and updates)	>= 418.39	>= 418.96
CUDA 10.0.130	>= 410.48	>= 411.31
CUDA 9.2 (9.2.148 Update 1)	>= 396.37	>= 398.26
CUDA 9.2 (9.2.88)	>= 396.26	>= 397.44
CUDA 9.1 (9.1.85)	>= 390.46	>= 391.29
CUDA 9.0 (9.0.76)	>= 384.81	>= 385.54
CUDA 8.0 (8.0.61 GA2)	>= 375.26	>= 376.51
CUDA 8.0 (8.0.44)	>= 367.48	>= 369.30
CUDA 7.5 (7.5.16)	>= 352.31	>= 353.66
CUDA 7.0 (7.0.28)	>= 346.46	>= 347.62

Thanks for the contribution of all the TAs.