

# Environment setup for tutorial examples

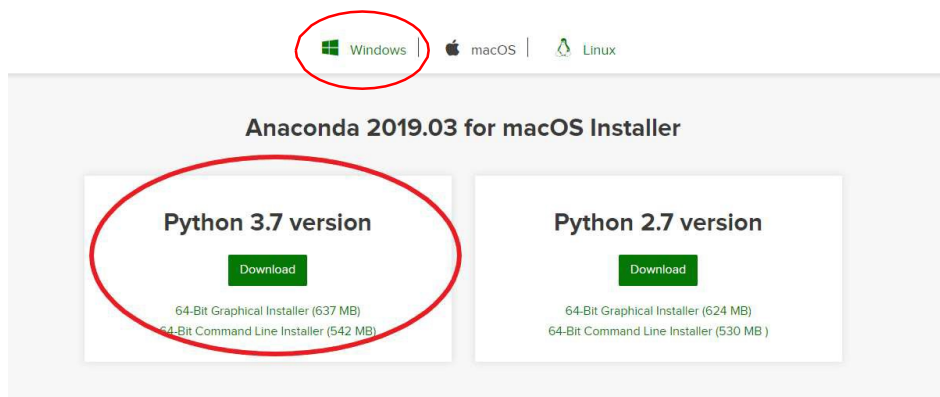
Dapeng Feng and Kuai Fang, Penn State Univ

## 1. Install Anaconda

<https://www.anaconda.com/distribution/>

We use python 3.7 version. Your choice of OS may differ. PyTorch and our code supports all major OSs including Linux, Mac, and Windows.

- Manage libraries, dependencies, and environments with Conda
- Develop and train machine learning and deep learning models with scikit-learn, TensorFlow, and Theano
- Analyze data with scalability and performance with Dask, NumPy, pandas, and Numba
- Visualize results with Matplotlib, Bokeh, Dashader, and Holoviews



## 2. Install pytorch

<https://pytorch.org/>. Select the correct OS.

QUICK START  
LOCALLY

Select your preferences and run the install command. Stable represents the most currently tested and supported version of PyTorch 1.2. This should be suitable for many users. Preview is available if you want the latest, not fully tested and supported, 1.2 builds that are generated nightly. Please ensure that you have met the prerequisites below (e.g., numpy), depending on your package manager. Anaconda is our recommended package manager since it installs all dependencies. You can also install previous versions of PyTorch. Note that LibTorch is only available for C++.

PyTorch Build	Stable (1.2)		Preview (Nightly)		
Your OS	Linux	Mac	Windows		
Package	Conda	Pip	LibTorch	Source	
Language	Python 2.7	Python 3.5	Python 3.6	Python 3.7	C++
CUDA	9.2		10.0	None	

Run this Command:

```
conda install pytorch torchvision cpuonly -c pytorch
```

There is a quick installation instruction on this website. Choose your option based on your OS and GPU. If you work with windows and don't have a GPU as this example suggests, simply run this in Anaconda Prompt or your shell:

```
conda install pytorch torchvision cpuonly -c pytorch
```

If you do have a GPU on the machine, it is absolutely beneficial to select the correct CUDA ver.

### 3. Install basemap package for visualization

Run this command in Anaconda Prompt or your shell:

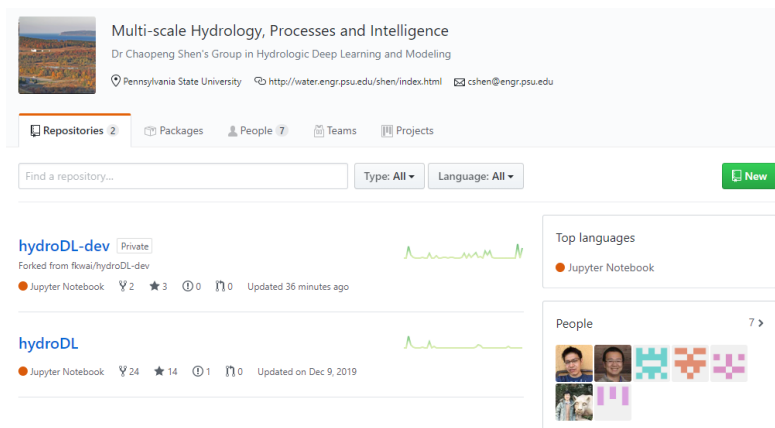
```
conda install -c anaconda basemap
```

**Notice:** here the basemap package may have some problems working with conda environment. You may need to change one line of our code before running and importing basemap package. This will be further introduced in step 5.

### 4. Get our repo

Install git: <https://www.atlassian.com/git/tutorials/install-git>

Fork and git clone our repo from these links:



**Public version (hydroDL):**

<https://github.com/mhpi/hydroDL>

`git clone https://github.com/mhpi/hydroDL.git`

**Development version (hydroDL-dev):**

<https://github.com/mhpi/hydroDL-dev>

`git clone https://github.com/mhpi/hydroDL-dev.git`

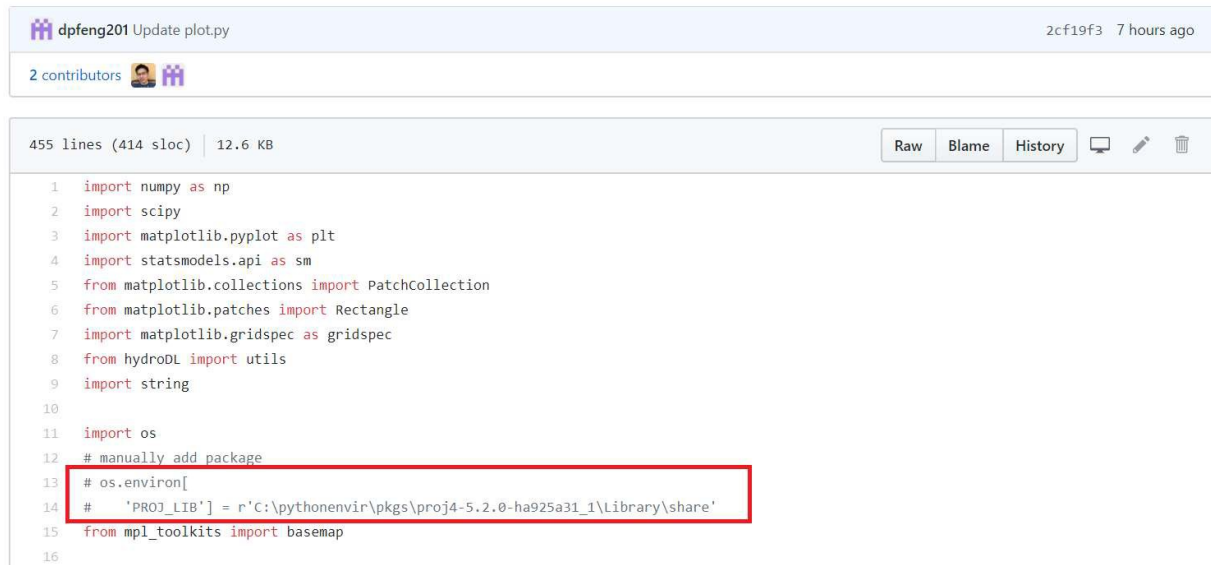
The public repo will be updated periodically as new papers approach acceptance.

## 5. address potential compatibility issue with the basemap visualization tool

In the past we have run into a compatibility issue with the basemap package. Since November 2019 it seems to have been addressed by 3<sup>rd</sup> party package updates. However, we detail the method (between === separators) to avoid this problem here in case it appears again in the future.

=====

The file that potentially needs to be modified is here: [{Repo}/hydroDL/post/plot.py](#)



```
1 import numpy as np
2 import scipy
3 import matplotlib.pyplot as plt
4 import statsmodels.api as sm
5 from matplotlib.collections import PatchCollection
6 from matplotlib.patches import Rectangle
7 import matplotlib.gridspec as gridspec
8 from hydroDL import utils
9 import string
10
11 import os
12 # manually add package
13 # os.environ[
14 #     'PROJ_LIB'] = r'C:\pythonenvir\pkgs\proj4-5.2.0-ha925a31_1\Library\share'
15 from mpl_toolkits import basemap
16
```

The code commented out on line 13 and 14 is for manually importing the basemap package. First check if you can successfully do “**from mpl\_toolkits import basemap**” without any error in python. To test this, you can simply type command ***python -c "from mpl\_toolkits import basemap"*** in your Anaconda Prompt or your shell. If no error emerges, you are all set. Nothing needs to be done. Go to the next steps.

Otherwise, if an error named “ **KeyError: 'PROJ\_LIB'** ” happens, you need to uncomment line 13 and 14, and also do the following modification. For the **line 14** please modify the path to the “share” folder on the right based on the “proj4” package location of your local computer. It should be: “**your\path\to\Anaconda3\pkgs\proj4-xxxxxxx\Library\share**” for Windows. Linux may be a little different but there should be a “share” folder under “proj4-xxx” package folder. Then you can try to run our example code in the following step.

=====

## 6. Run the example code

After setting up the environment, you can try to run the examples listed in the home page of our repo. They are either Jupyter notebook or .py files. Jupyter notebook is installed with Anaconda already. The Jupyter Notebook App can be launched by clicking on the Jupyter Notebook icon installed by Anaconda in the start menu (Windows) or by typing in Anaconda Prompt or cmd on Windows:

*jupyter notebook*

In Linux you will need to type in the terminal:

*jupyter-notebook*