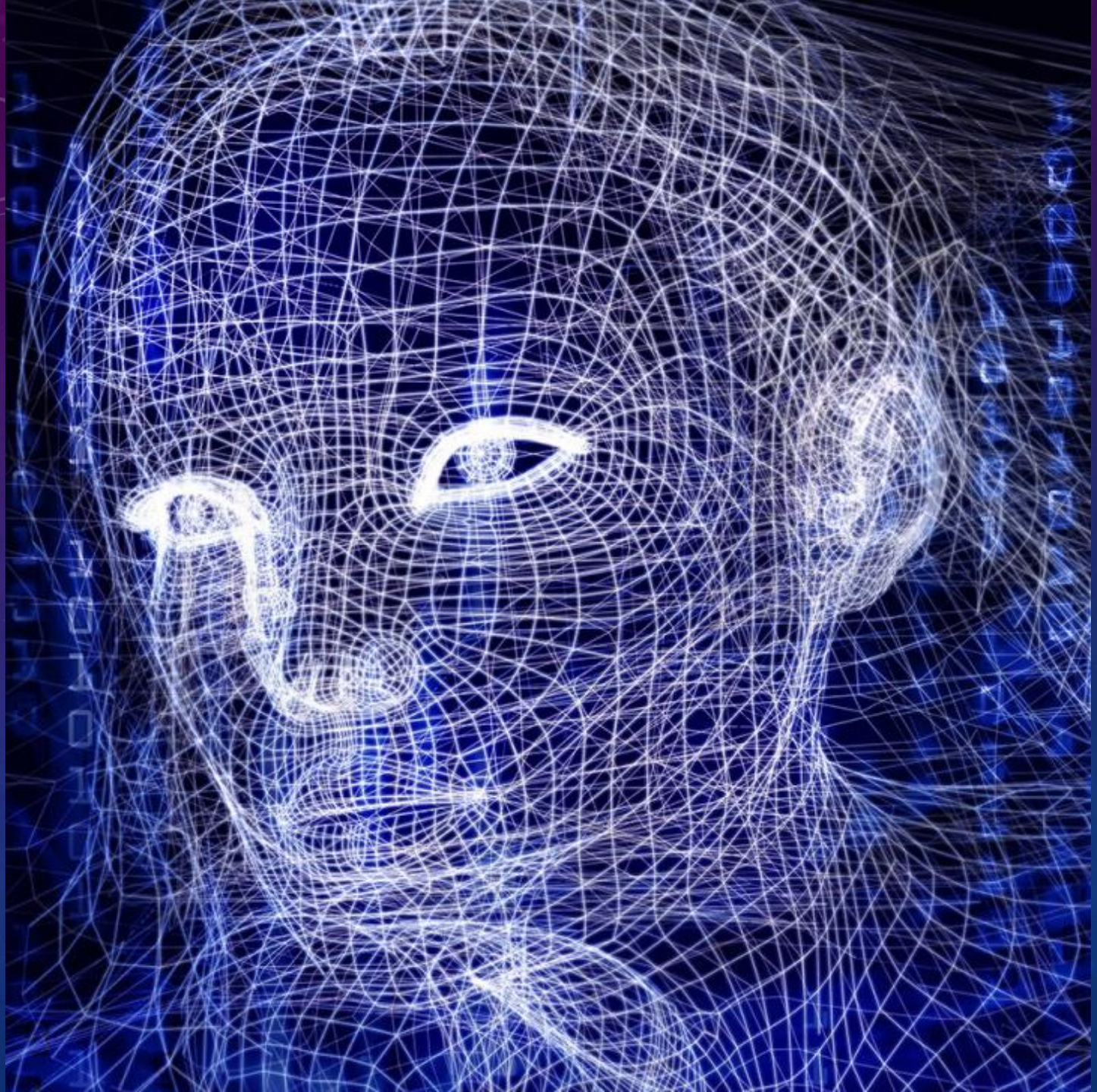


*COMPUTER VISION AND
ITS APPLICATIONS
CH 1_EXERCISE*

徐繼聖

Gee-Sern Jison Hsu

National Taiwan University of Science
and Technology



What is Pytorch?

- It's a Python-based scientific computing package targeted at two sets of audiences:
 - A replacement for NumPy to use the power of GPUs
 - a deep learning research platform that provides maximum flexibility and speed



Installation Process

1. Choose the pytorch version

Pytorch download link : <https://pytorch.org/get-started/locally/>

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
CUDA	9.2	10.0	None	

Run this Command:

```
conda install pytorch torchvision cudatoolkit=10.0 -c pytorch
```

Run this code in the Terminal

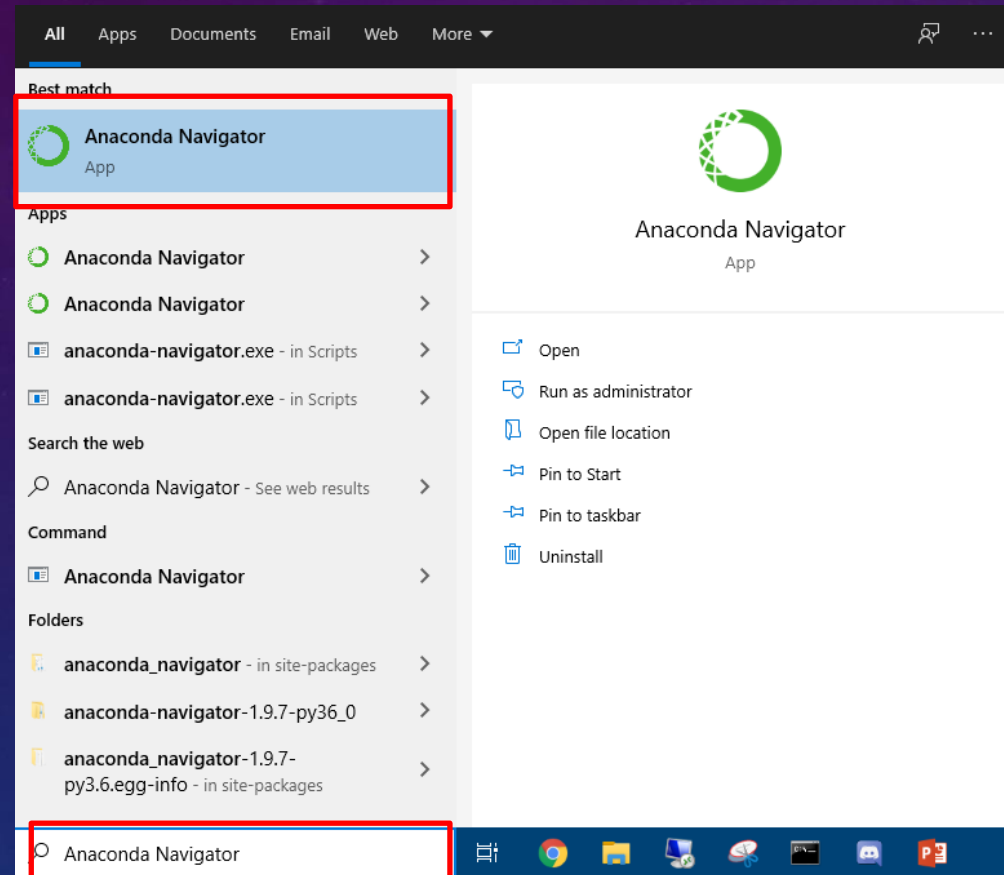
If you just want to use cpu, you need to select None

Installation Process

- Pytorch 1.2:
 - `conda install pytorch torchvision cudatoolkit=9.2 -c pytorch -c defaults -c numba/label/dev`
- Because Pytorch(1.2) is only compatible with CUDA 9.2 and 10, the computers in the PC room are not compatible.
- Instead please use the following command to install the Pytorch(1.0, Cuda 8)
 - `conda install pytorch torchvision cudatoolkit=8.0 -c pytorch`

Installation Process

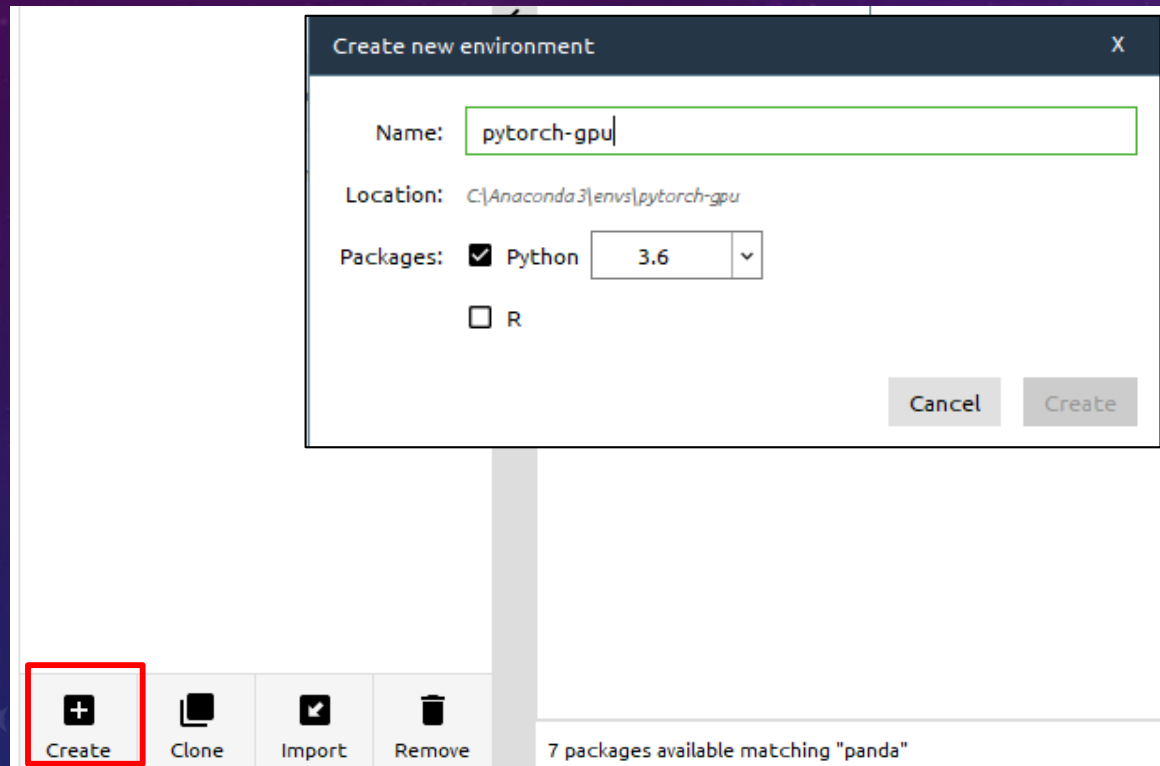
2. Open “Anaconda Navigator”



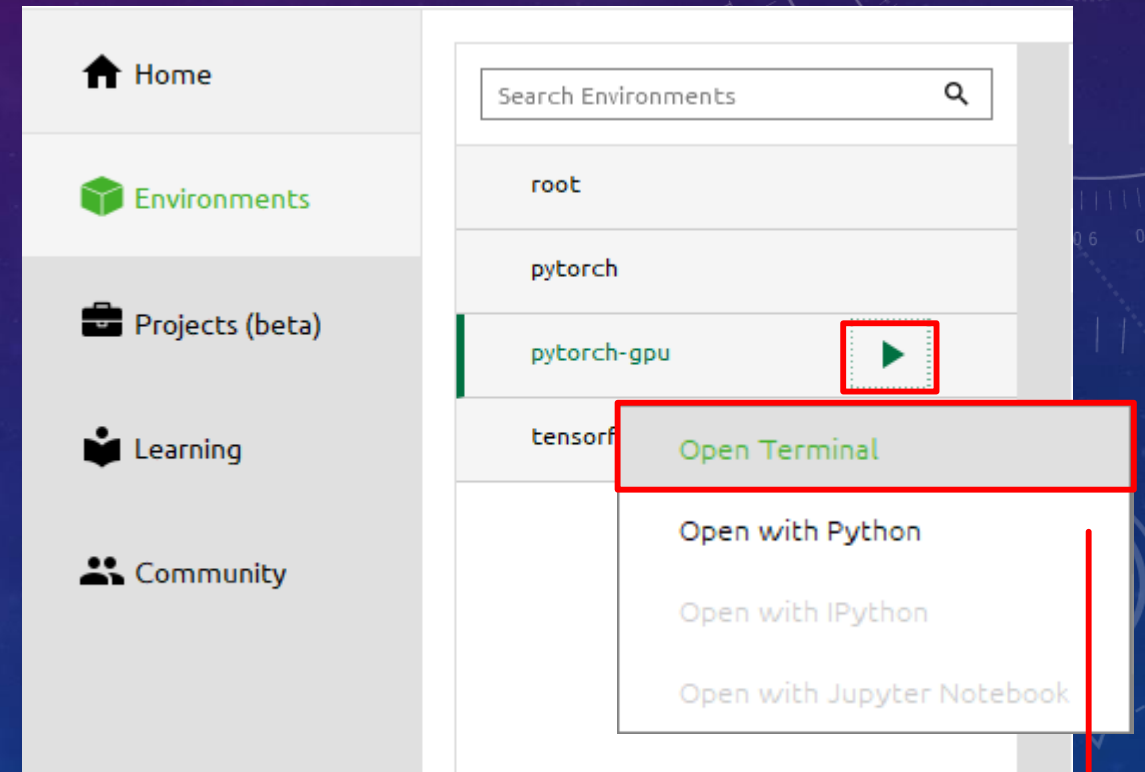
Type “Anaconda Navigator” to search

Installation Process

3. Create the Pytorch environment



Click the Create



Open the Terminal

Installation Process

4. Install Pytorch

```
C:\Windows\system32\cmd.exe

(C:\Anaconda3\envs\pytorch-gpu) C:\Users\c_user>conda install pytorch torchvision cudatoolkit=8.0 -c pytorch
Fetching package metadata .....
Solving package specifications: .

Package plan for installation in environment C:\Anaconda3\envs\pytorch-gpu:

The following NEW packages will be INSTALLED:

blas:          1.0-mkl
cffi:          1.12.2-py36h7a1dbcf_1
cudatoolkit:   8.0-4
freetype:      2.9.1-ha9979f8_1
icc_rt:        2019.0.0-h0cc432a_1
```

5. Test Pytorch

```
(C:\Users\Micky\Anaconda3\envs\pytorch) C:\Users\Micky>python
Python 3.6.6 |Anaconda, Inc.| (default, Jun 28 2018, 11:27:44) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import torch
>>> x = torch.rand(5,3)
>>> print(x)
tensor([[0.8665, 0.1280, 0.1916],
        [0.7790, 0.9906, 0.6345],
        [0.9891, 0.2146, 0.1391],
        [0.2996, 0.3896, 0.9302],
        [0.0182, 0.9602, 0.5271]])
>>> torch.cuda.is_available()
True
```

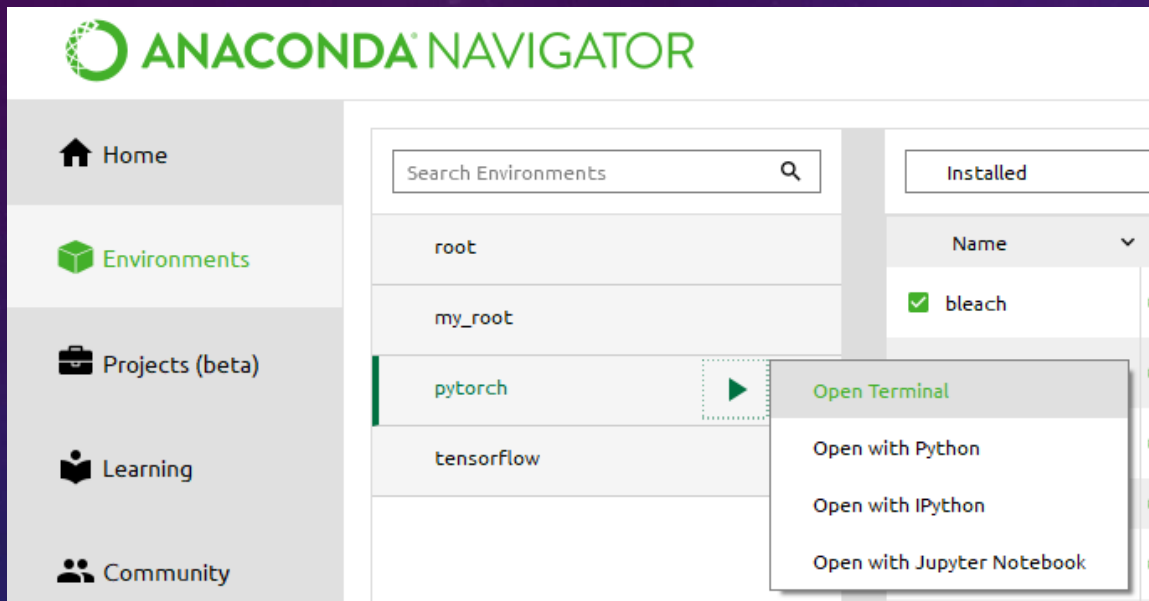
Enter python

This shows the installation is completed

Print "True" means GPU and CUDA can support the Pytorch

Use The Python File

- Step 1: Download the file from Moodle and save in the D: Drive
- Step 2: Open the anaconda-navigator and choose your environment
- Step 3: Open the terminal



- Step 4: open the python file

```
C:\Windows\system32\cmd.exe
(C:\ProgramData\Anaconda3\envs\pytorch) C:\Users\c_user>D:
(C:\ProgramData\Anaconda3\envs\pytorch) D:\>python exercisel_linear_regression.py
```

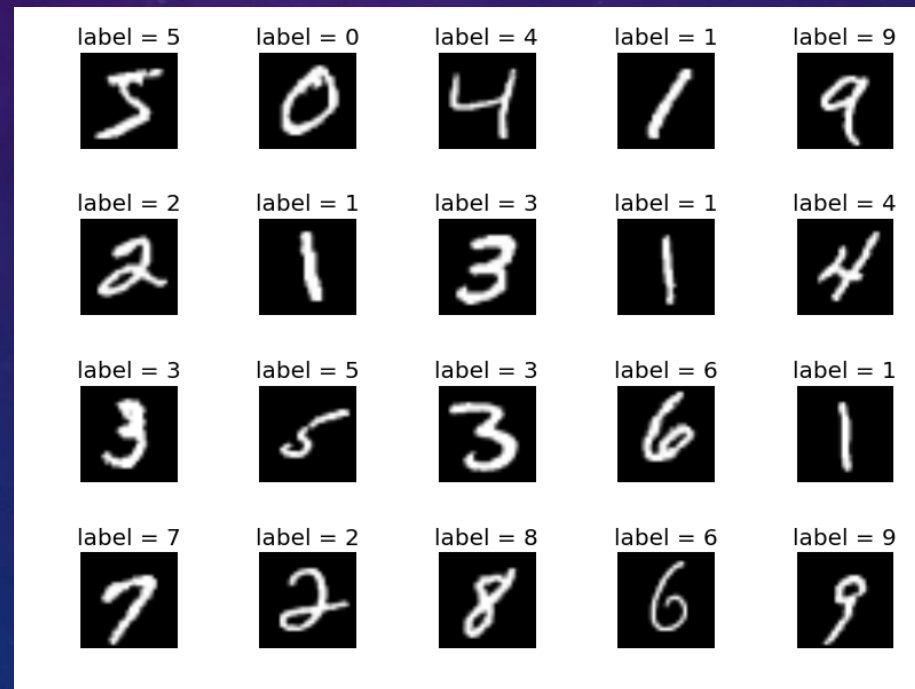
Cd changes the current working directory,
example: (cwd)

cd D:\James

cwd is then inside the James folder

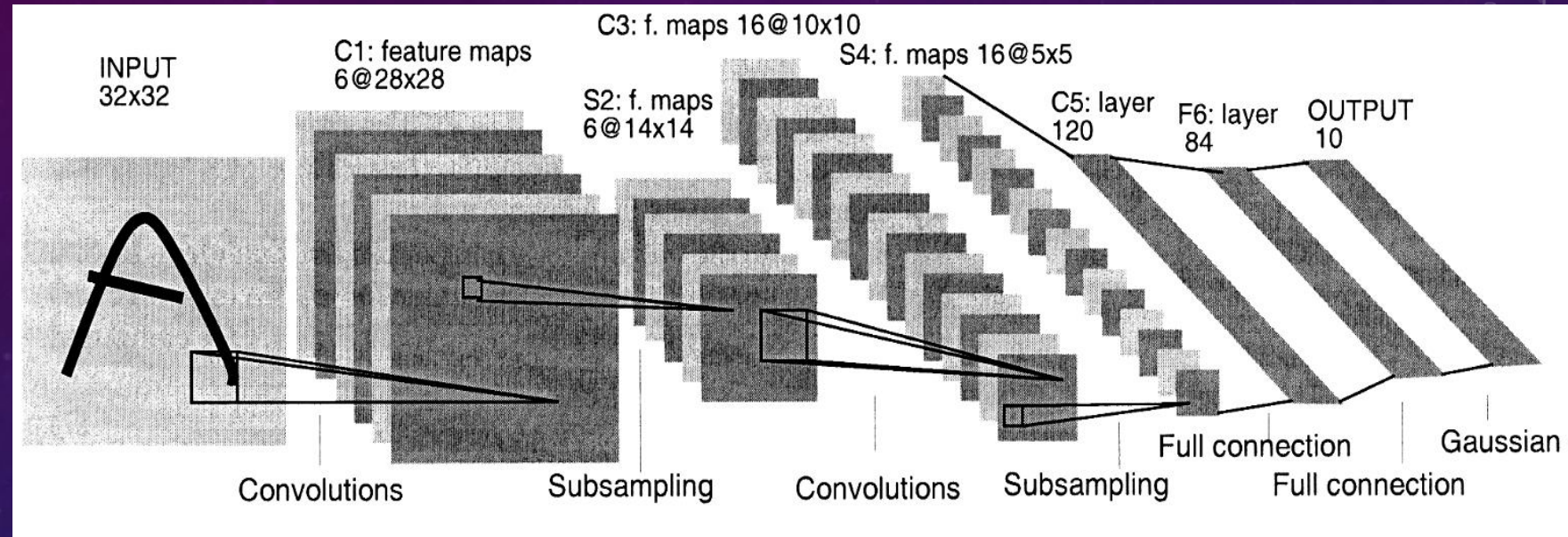
Example 1: Training LeNet on Mnist

- Mnist database is dataset for handwritten digits.
- The dataset consists of pair, “handwritten digit image” and “label”.
 - handwritten digit image: This is gray scale image with size 28 x 28 pixel.
 - label : This is actual digit number this handwritten digit image represents. It is either 0 to 9.



Sample

Example 1: Training LeNet on Mnist



```
LeNet(  
  (conv1): Conv2d(1, 6, kernel_size=(5, 5), stride=(1, 1))  
  (conv2): Conv2d(6, 16, kernel_size=(5, 5), stride=(1, 1))  
  (fc1): Linear(in_features=400, out_features=120, bias=True)  
  (fc2): Linear(in_features=120, out_features=84, bias=True)  
  (fc3): Linear(in_features=84, out_features=10, bias=True)  
)
```

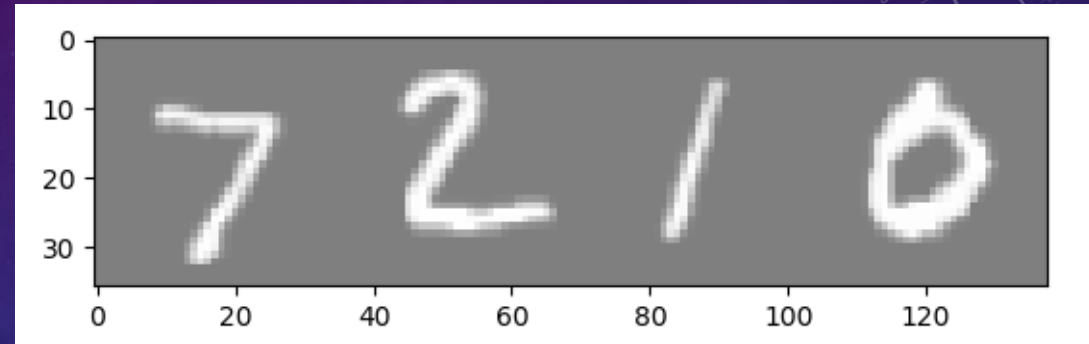
10 classes (0 to 9)

Example 1: Training LeNet on Mnist

The samples in Mnist :



Result :



```
Epoch : 1 steps : 2000 Training Loss : 1.643718808606267
Epoch : 1 steps : 4000 Training Loss : 0.27060963289509526
Epoch : 1 steps : 6000 Training Loss : 0.15463385406913585
Epoch : 2 steps : 2000 Training Loss : 0.1041009760457091
Epoch : 2 steps : 4000 Training Loss : 0.09052719001102014
Epoch : 2 steps : 6000 Training Loss : 0.07582578212058434
Finished Training
GroundTruth:      7      2      1      0
Predicted:        7      2      1      0
Accuracy : 98 %
```


Exercise 1: Result on MNIST

- (a) Please download “Exercise1_Mnist.7z” from Moodle.
- (b) Run the “Exercise1_Mnist.py” to train the model.
- (c) Get familiar with Python and Pytorch.

Please copy your results and code and paste to a MS Word, then upload to Moodle.