# Lab 12: Deep Learning Tutorial 2

By Sam Liu, Allen Ou, last modified on 5/5 2020.

The objective of this tutorial is to introduce to the basic concept of DL, include the MLP model, forward/backward propagation. We also prepared some example code for basic Pytorch libraries in Jupyter notebook.

Why deep?

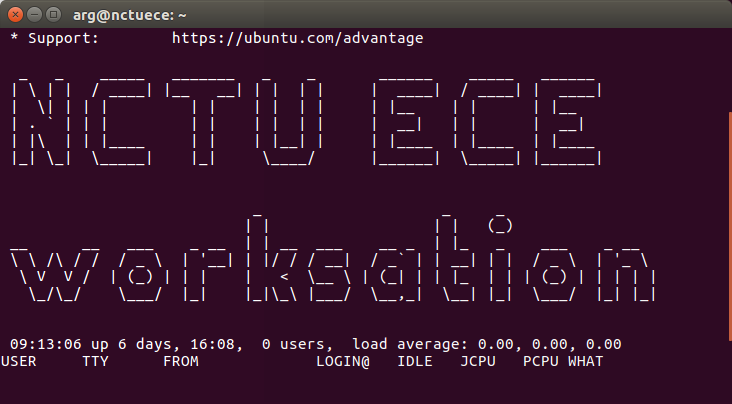
## Hardware and Software Setup

We have already installed ROS in the VirtualBox image we provided, for those who use native Ubuntu, please follow the step [here](http://wiki.ros.org/kinetic/Installation/Ubuntu).

Access GPU machine with ssh

**laptop $ ssh [username]@140.113.148.xxx**

type the passward then you will see like this

-

**ws : workstation**

Clone the course repository

**ws $ cd  
ws $  
git clone** [**https://github.com/Sensing-Intelligent-System/sis\_lab\_all\_2020**](https://github.com/Sensing-Intelligent-System/sis_lab_all_2020) **-b [your\_branch]**

**ws $ cd sis\_lab\_all\_2020 && git pull origin master**

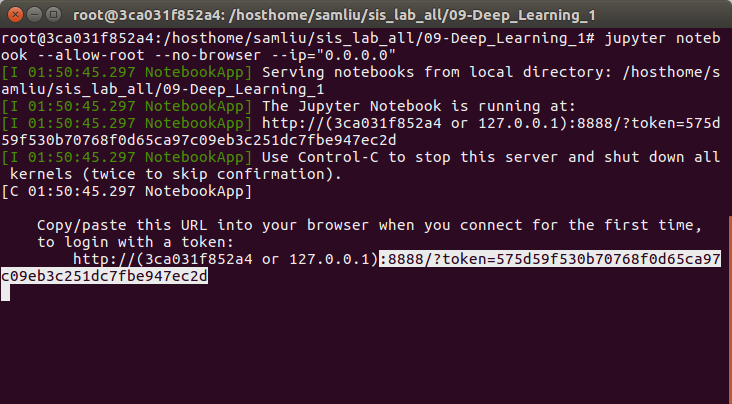
Build dockerfile

**ws $ cd ~/sis\_lab\_all\_2020/12-Deep\_Learning\_2/Dockerfiles && source docker\_build.sh**

Run the docker and jupyter notebook

**ws $ cd .. && source docker\_run.sh**

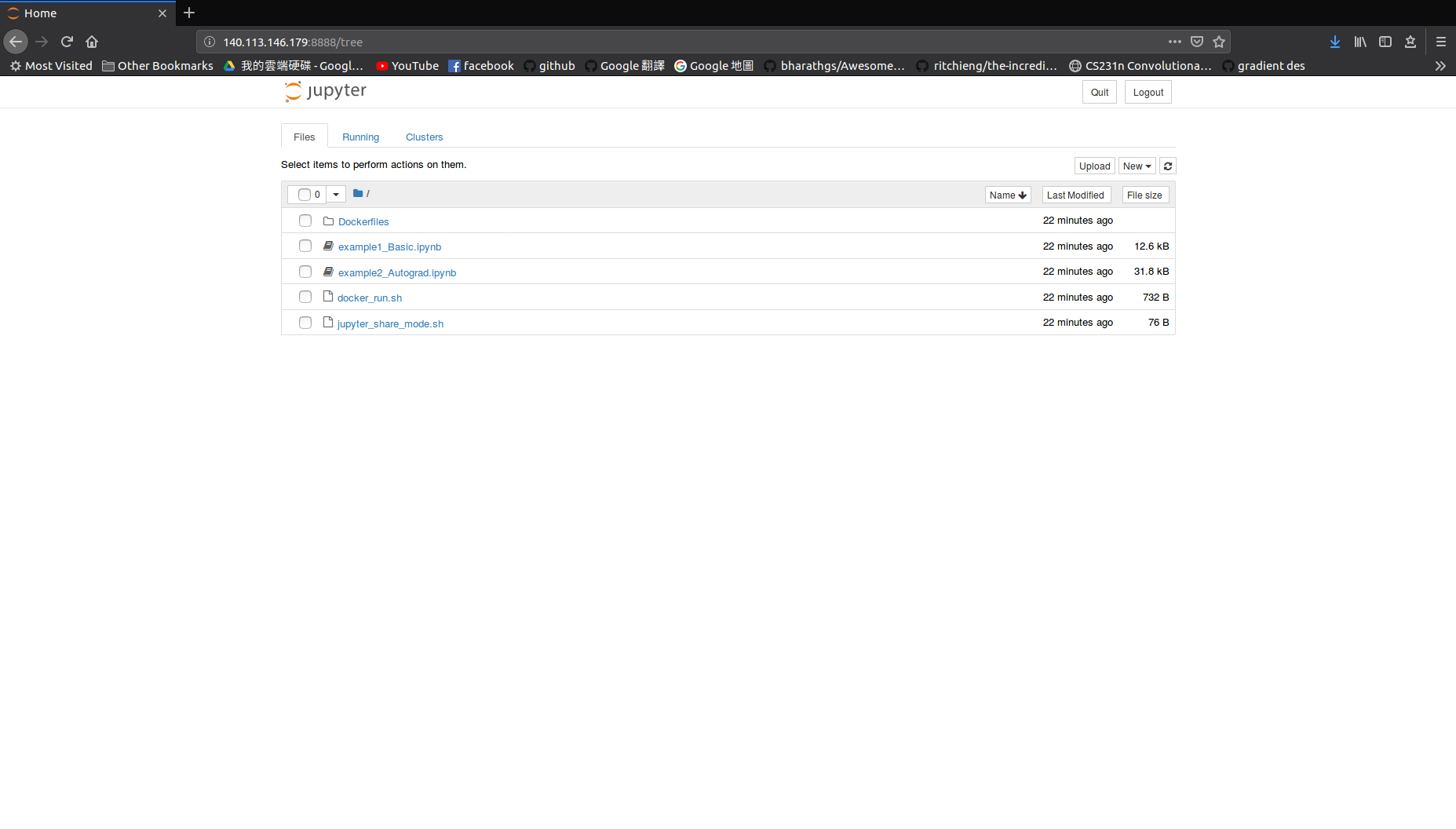
**container $ source jupyter\_no\_broswer.sh**

****

e.g. http://**[workstation\_ip]**:**[port]**?token=**[xxxxx]**

Turn on the web browser on local and type the **workstation’s IP** and **token** from above

You will see something like this



[**Lab 12: Deep Learning Tutorial 2**](#_74gram21yqax) **1**

[Hardware and Software Setup](#_chc57ow4x0s1) 1

[Overview](#_x5puhc1p2jjo) 3

[Usage of jupyter notebook](#_hqm01wfyfbgo) 3

[Topics and Activities](#_g0fw0bgwxhdt) 4

[Topic/Activity 1 Introduction to CNN](#_aadm982kyolo) 4

[Topic 1.1 Pytorch Layers of CNN](#_rl15slrm10vt) 4

[Topic/Activity 2 Prediction with a pretrained model](#_sf5mhl8ztpdm) 4

[Topic 2.1 Pytorch Model saving and loading](#_v4sjn3hb2lab) 4

[**Assignment.**](#_p0x456r0carb) **5**

[Reference](#_uwwglharlfb) 6

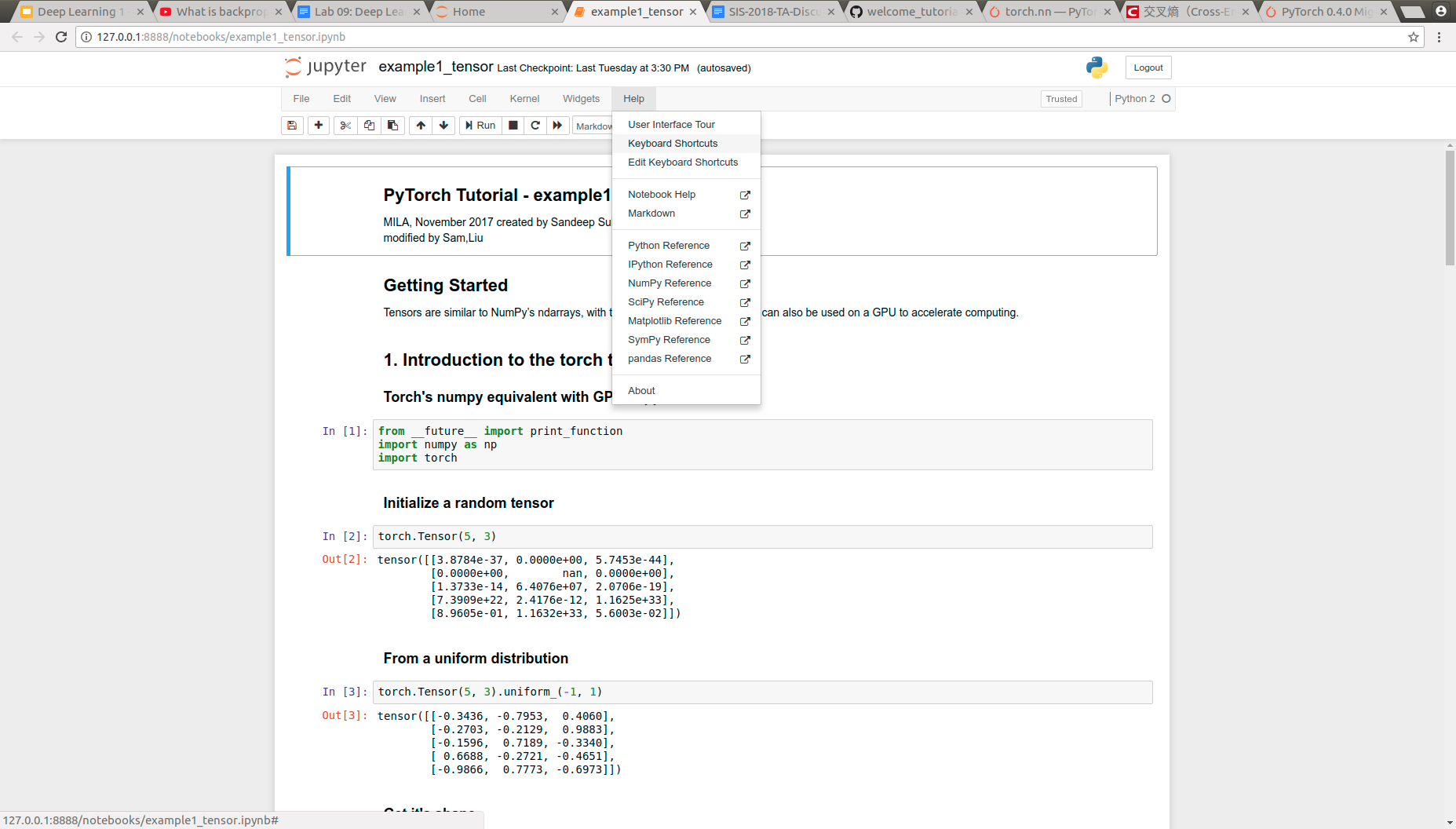
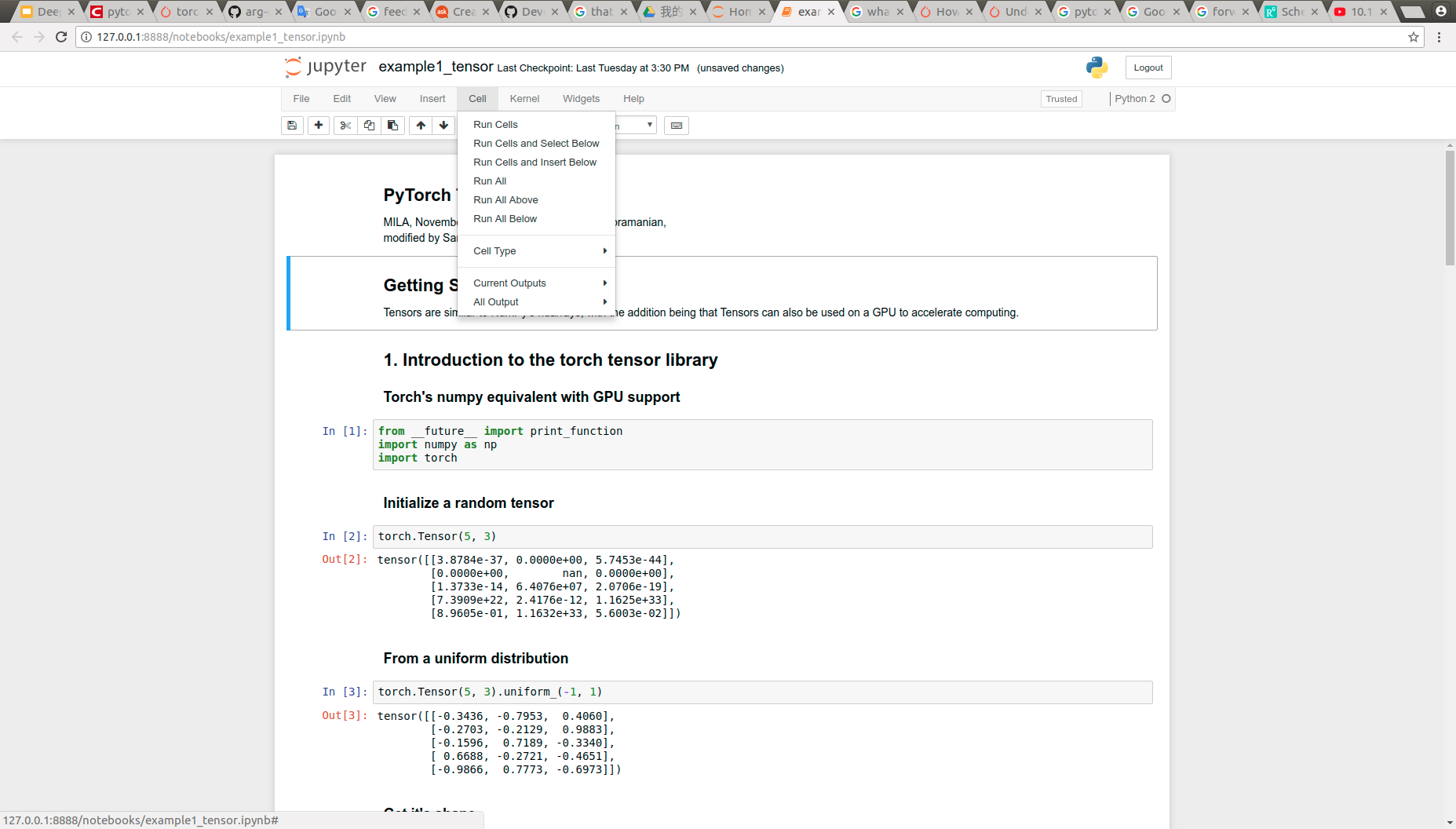
## Overview

Estimated Time to Finish: 1.5 hours

After completing this tutorial you should

* understand the basic deep learning architecture, concept of forward/backward propagation.
* understand how to use Pytorch module to implement the regression model.

## Usage of jupyter notebook

1. Keyboard shortcuts: (Help → Keyboard shortcuts)  
   
2. Run cell: [Shift+Enter]
3. 

## Topics and Activities

### Topic/Activity 1 Introduction to CNN

#### **Topic 1.1 Pytorch Layers of CNN**

Turn on jupyter notebook, and click **example1\_layers.ipynb**

(Estimated Time to Finish: 10 minutes)

### Topic/Activity 2 Prediction with a pretrained model

#### **Topic 2.1 Pytorch Model saving and loading**

Turn on jupyter notebook, and click **example2\_saving\_loading\_models.ipynb**

(Estimated Time to Finish: 15 minutes)

**Disscussion:**

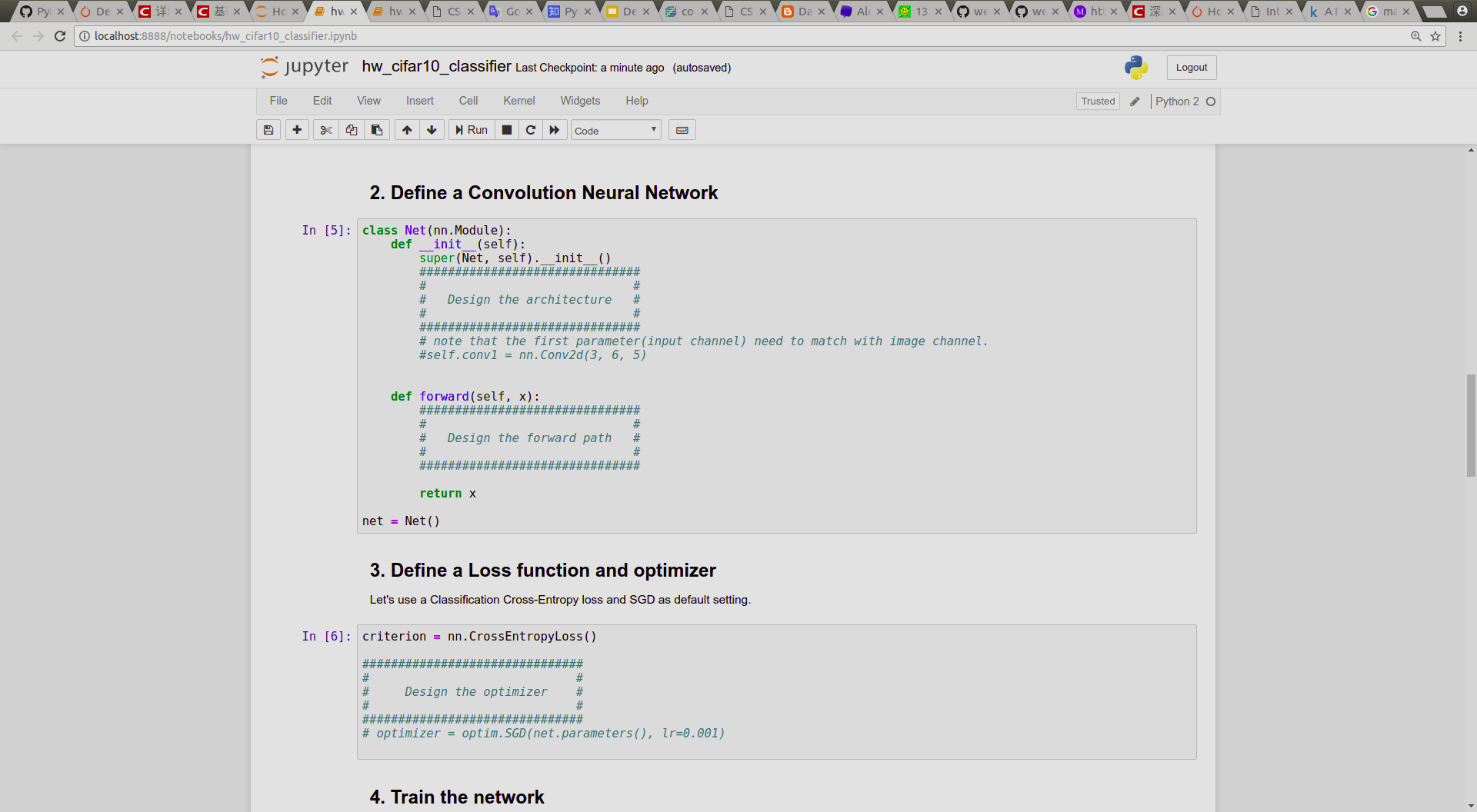
1. **Could you describe how does the convolution operation work for feature extraction?**
2. **What's the advantage and drawback of ‘ReLu’ function**
3. **Why do we choose the Softmax or LogSoftmax to be a output layer in classification problem？**
4. **In example3, why the ‘state dictionary loading’ is recommended way to load a pretrained model? Compare with ‘entire model loading’ (no standard answer)**

# 

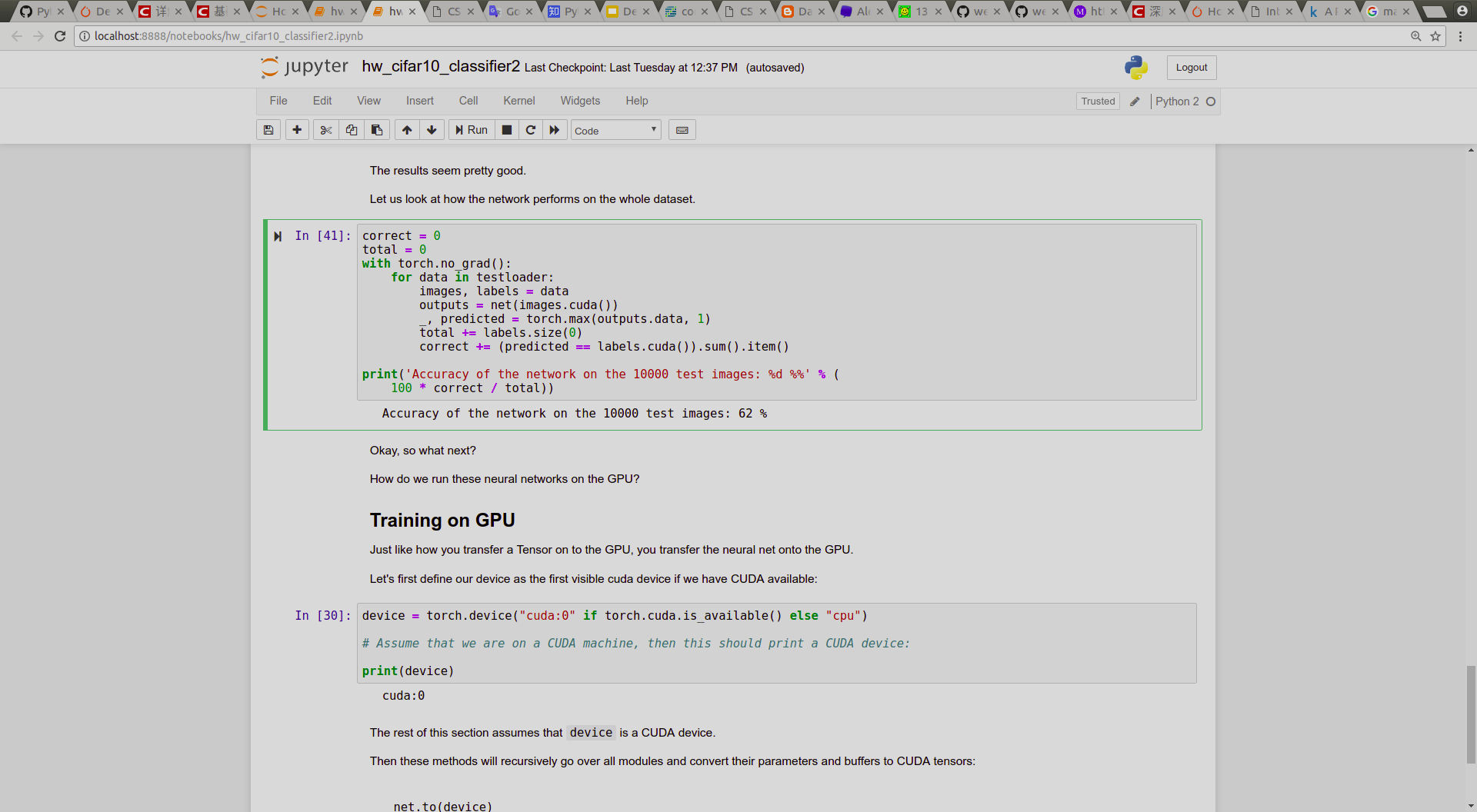
# 

# Assignment.

In **hw\_cifar10\_classifier.ipynb** we provide incomplete code to ask students to complete, the parts that need to be designed are as follows.



**requirement:  
1. The accuracy need to be higher than 60%, and take a screenshot for the result.**

****

## Reference

Pytorch official tutorial: <https://pytorch.org/tutorials/>

Mila-udem pytorch tutorial: <https://github.com/mila-udem/welcome_tutorials/tree/master/pytorch>