Introduction to Deep Learning (CS474)

Lecture 1

Rupam

Outline

- Brief overview of Pytorch.
- Popular Deep Learning Frameworks.
- Installing Pytorch in your machine.
- Introduction to Google Colab.
- Getting started with Google Colab.

Rupam

Brief overview of 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.

Rupam

Popular Deep Learning Frameworks

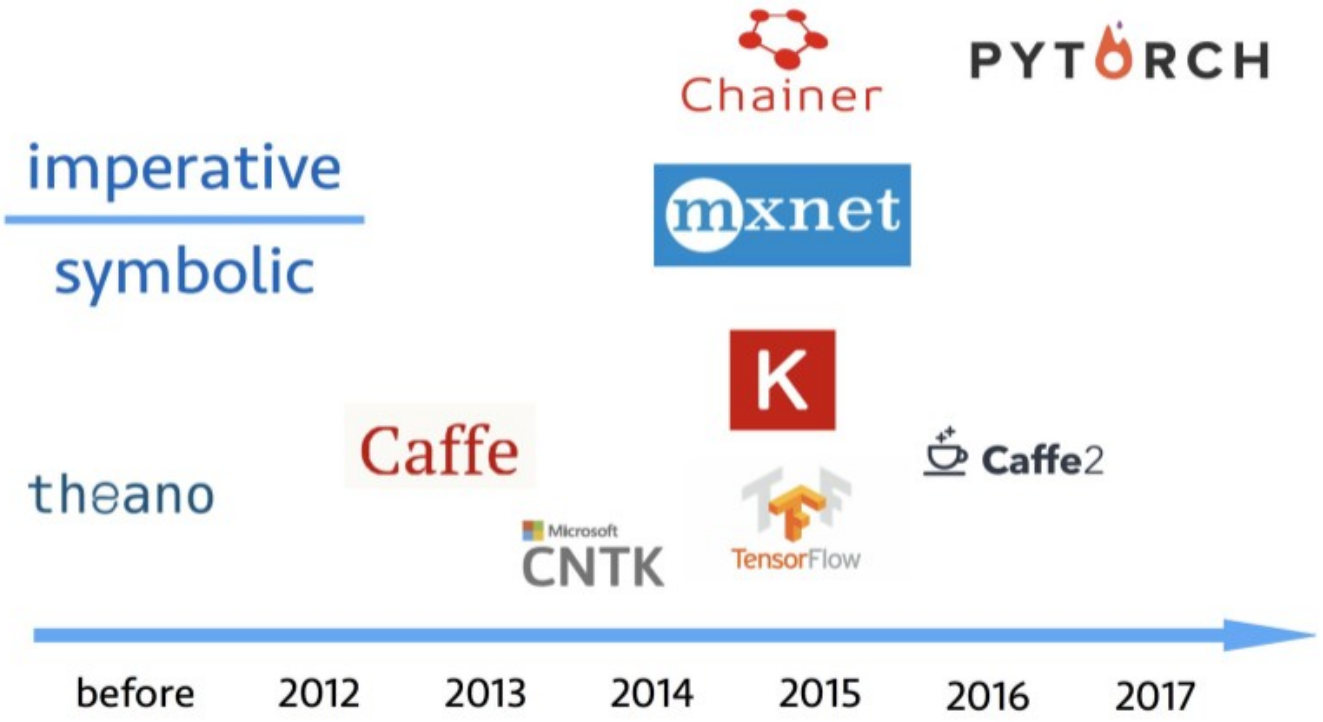


Image credit: Gluon: new MXNet interface to accelerate research

Rupam

Popular Deep Learning Frameworks

Imperative: Imperative-style programs perform computation as you run them

```
import numpy as np
a = np.ones(10)
b = np.ones(10) * 2
c = b * a
d = c + 1
```

Symbolic: define the function first, then compile them

```
A = Variable('A')
B = Variable('B')
C = B * A
D = C + Constant(1)
# compiles the function
f = compile(D)
d = f(A=np.ones(10), B=np.ones(10)*2)
```

Image credit: Gluon: new MXNet interface to accelerate research

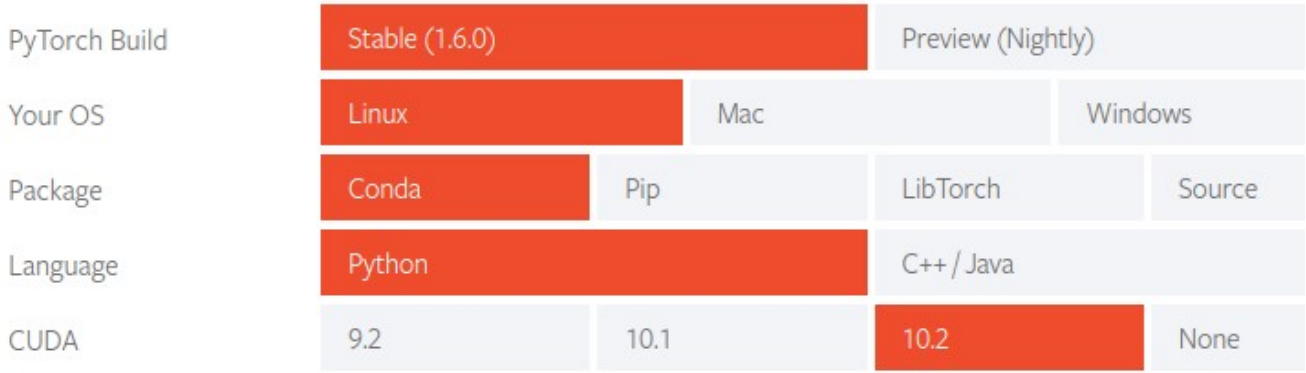
Rupam

Installing Pytorch in your machine

Let's start locally!

- I am assuming that
  - you have installed python 3.5.
  - Your OS is Ubuntu!
- You may look into the following figure along with the following URL.

  https://pytorch.org/get-started/locally/



Rupam

Installing Pytorch in your machine

- To install the PyTorch binaries, you will need to use one of two supported package managers: Anaconda or pip.

- To install Anaconda, you will use the command-line installer. Right-click on the 64-bit installer link, select Copy Link Location, and then use the following commands:

```
# The version of Anaconda may be different depending on when you are installing`
curl -O https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
sh Miniconda3-latest-Linux-x86_64.sh
# and follow the prompts. The defaults are generally good.`
```

> *You may have to open a new terminal or re-source your `~/.bashrc` to get access to the `conda` command.*

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

Rupam

Installing Pytorch in your machine

- Verification

```
(base) rupam-iiitbh@rupamiiitbh-HP-348-G4:~$ mkdir pytorch_projects
(base) rupam-iiitbh@rupamiiitbh-HP-348-G4:~$ cd pytorch_projects/
(base) rupam-iiitbh@rupamiiitbh-HP-348-G4:~/pytorch_projects$ mkdir basic
(base) rupam-iiitbh@rupamiiitbh-HP-348-G4:~/pytorch_projects$ cd basic/
(base) rupam-iiitbh@rupamiiitbh-HP-348-G4:~/pytorch_projects/basic$ gedit test_torch.py
(base) rupam-iiitbh@rupamiiitbh-HP-348-G4:~/pytorch_projects/basic$ python3 test_torch.py
tensor([[0.4025, 0.9654, 0.7734],
        [0.2092, 0.0662, 0.0105],
        [0.1729, 0.1292, 0.8992],
        [0.3075, 0.3552, 0.5083],
        [0.3234, 0.7774, 0.3524]])
(base) rupam-iiitbh@rupamiiitbh-HP-348-G4:~/pytorch_projects/basic$
```

- **test_torch.py** contain following lines of code which will construct a randomly initialized tensor.

```
from __future__ import print_function
import torch
x = torch.rand(5, 3)
print(x)
```

Rupam

Installing Pytorch in your machine

- Now, we have installed pytorch correctly.
  - This demonstration don't showcase the GPU utilization.
  - PyTorch utilize CUDA for fast processing which we have not demonstrated till now.
  - The NVIDIA CUDA Toolkit provides a development environment for creating high performance GPU-accelerated applications.

Rupam

Introduction to Google Colab.

- Free GPUs for Everyone!

- **Google Colaboratory** is a free online cloud-based Jupyter notebook environment that allows us to train our machine learning and deep learning models on CPUs, GPUs, and TPUs.

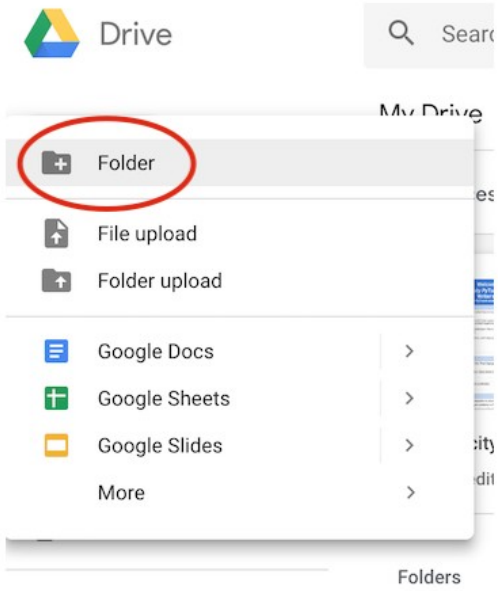
- Colab gives us 12 hours of continuous execution time. After that, the whole virtual machine is cleared and we have to start again.

Slide Credit: Analytics Vidhya

Rupam

Getting started with Google Colab.

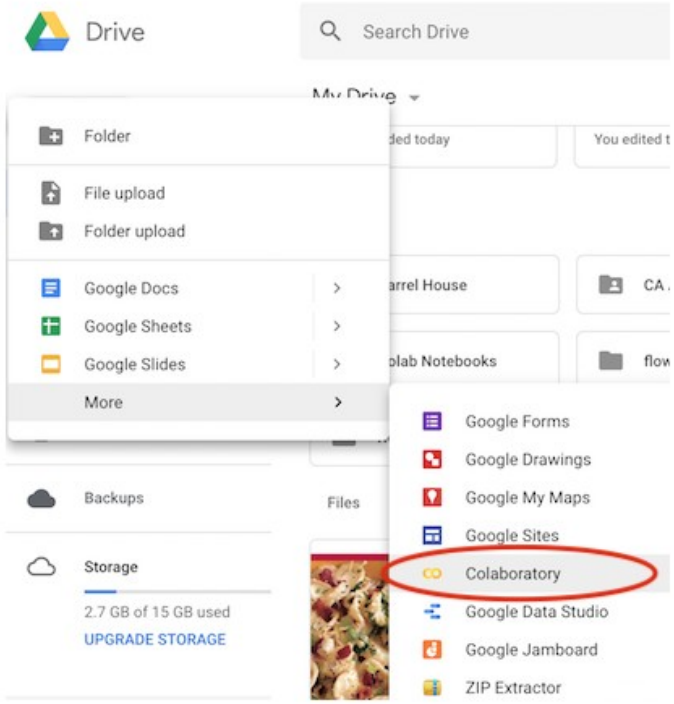
- Setting up your Google drive.
  - Create a folder for your Jupyter notebooks.



Rupam

Getting started with Google Colab.

- Creation of a new **notebook** within that *folder*.

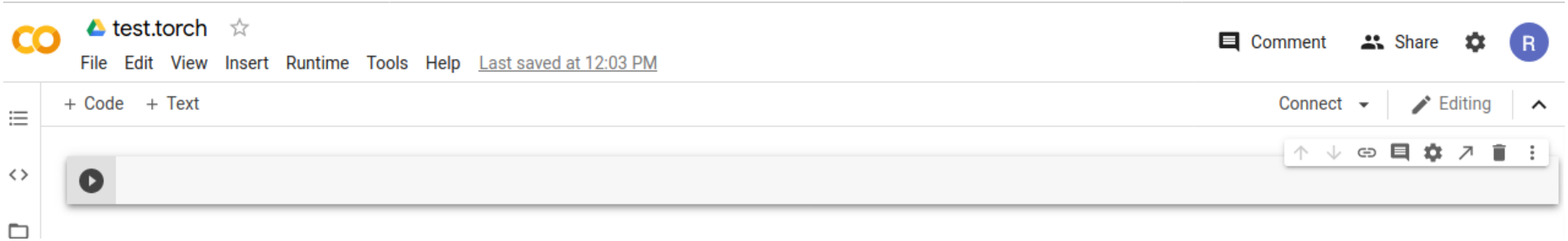


Slide credit: Anne Bonner

Rupam

Getting started with Google Colab.

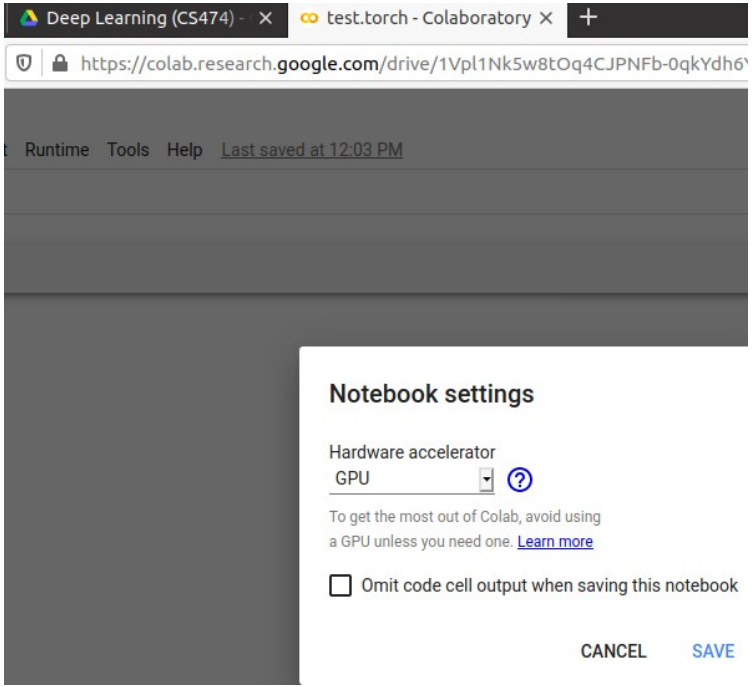
- Rename the notebook to “test_torch”present within that *folder*.
- Open that file in Google Collaboratory. You may observe the following things in your web browser.



Rupam

Getting started with Google Colab.

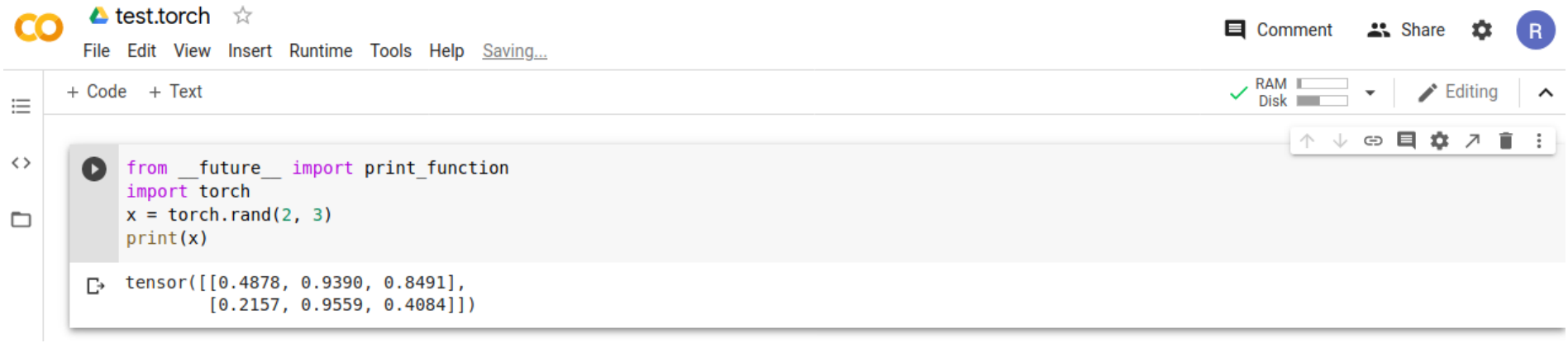
- Set up your GPU.
  - Go to the “runtime” drop down menu, selecting “**change runtime type**” and selecting GPU in the hardware accelerator drop-down menu!



Rupam

Getting started with Google Colab.

- Write your code within the *test_torch* notebook.



- Google *Colab* now supports native **Pytorch**.
- We run the the same code in Google colab which was executed in our *local* machine.

Rupam

Getting started with Google Colab.

- Now, we would like to perform matrix multiplication through our CPU and numpy!

Open

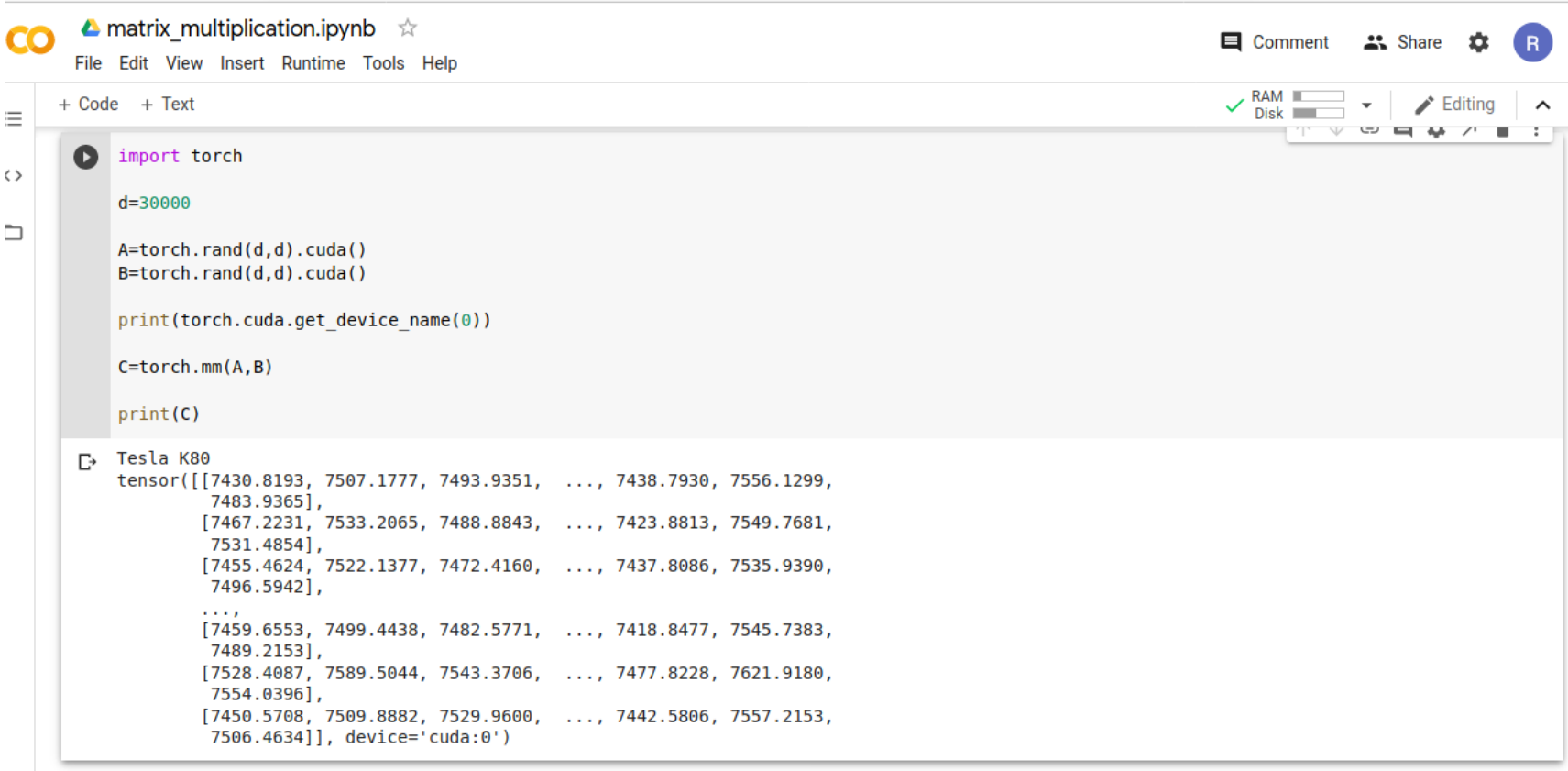
matrix_multiplication_numpy.py
~/pytorch_projects/basic

```
import numpy as np

d=30000

A=np.random.rand(d,d).astype(np.float32)
B=np.random.rand(d,d).astype(np.float32)

C=A.dot(B)

print(C)
```

Rupam

Getting started with Google Colab.

- Now, we would like to perform matrix multiplication through GPU and torch!



References

- All the contents present in the slides are taken from various online resources. These slides are used for *academic* purposes only.
  - https://pytorch.org/
  - https://developer.nvidia.com/cuda-toolkit