

CS5242 : Neural Networks and Deep Learning

Lecture 2: Linear Algebra and PyTorch

Semester 1 2021/22

Xavier Bresson

<https://twitter.com/xbresson>

Department of Computer Science
National University of Singapore (NUS)



Outline

- Review of linear algebra :
 - Vectors
 - Matrices
 - Matrix-vector multiplication
 - Matrix-matrix multiplication
 - Inner and outer products
 - Tensors
- PyTorch

Outline

- Review of linear algebra :
 - Vectors
 - Matrices
 - Matrix-vector multiplication
 - Matrix-matrix multiplication
 - Inner and outer products
 - Tensors
- PyTorch

Vectors

Vector Addition:

$$\begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} + \begin{bmatrix} 5 \\ -1 \\ 2 \end{bmatrix} = \begin{bmatrix} 6 \\ 0 \\ 4 \end{bmatrix}$$

Scalar Multiplication:

$$2 \cdot \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 2 \\ 2 \\ 4 \end{bmatrix}$$

Inner Product

- a.k.a. scalar product :

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \cdot \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = x_1y_1 + x_2y_2 + x_3y_3$$

Example :

$$\begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = 1 + 2 + 6 = 9$$

Inner Product

- Vector representation :

$$\vec{\mathbf{x}} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \quad \vec{\mathbf{y}} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix}$$

$$\vec{\mathbf{x}} \cdot \vec{\mathbf{y}} = x_1y_1 + x_2y_2 + x_3y_3 + x_4y_4$$

Outline

- Review of linear algebra :
 - Vectors
 - Matrices
 - Matrix-vector multiplication
 - Matrix-matrix multiplication
 - Inner and outer products
 - Tensors
- PyTorch

Matrices

$$W = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} \quad \text{is a } 3 \times 2 \text{ matrix}$$

Height/rows

Width/columns

$$W = \begin{bmatrix} w_{11} & w_{12} & w_{13} & w_{14} & w_{15} \\ w_{21} & w_{22} & w_{23} & w_{24} & w_{25} \\ w_{31} & w_{32} & w_{33} & w_{34} & w_{35} \end{bmatrix} \quad \text{is a } 3 \times 5 \text{ matrix}$$

Transpose of a matrix

$$W = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 5 & 0 & 5 & 0 \\ 2 & 2 & 2 & 2 & 2 \end{bmatrix} \quad \text{is a } 3 \times 5 \text{ matrix}$$

$$W^T = \begin{bmatrix} 1 & 0 & 2 \\ 1 & 5 & 2 \\ 1 & 0 & 2 \\ 1 & 5 & 2 \\ 1 & 0 & 2 \end{bmatrix} \quad \text{is a } 5 \times 3 \text{ matrix}$$

Matrix operations

Matrix Addition:

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} + \begin{bmatrix} 10 & 10 \\ 20 & 20 \\ 30 & 30 \end{bmatrix} = \begin{bmatrix} 11 & 12 \\ 23 & 24 \\ 35 & 36 \end{bmatrix}$$

Scalar multiplication:

$$2 \cdot \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} = \begin{bmatrix} 2 & 4 \\ 6 & 8 \\ 10 & 12 \end{bmatrix}$$

Outline

- Review of linear algebra :
 - Vectors
 - Matrices
 - Matrix-vector multiplication
 - Matrix-matrix multiplication
 - Inner and outer products
 - Tensors
- PyTorch

Matrix-Vector Multiplication

$$\begin{bmatrix} w_{11} & w_{12} & w_{13} & w_{14} & w_{15} \\ w_{21} & w_{22} & w_{23} & w_{24} & w_{25} \\ w_{31} & w_{32} & w_{33} & w_{34} & w_{35} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} (\text{row}_1) \cdot \vec{x} \\ (\text{row}_2) \cdot \vec{x} \\ (\text{row}_3) \cdot \vec{x} \end{bmatrix}$$

3 x 5 matrix

input vector
has 5 entries

output vector
has 3 entries

Example 1

$$W = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \quad \vec{x} = \begin{bmatrix} 1 \\ -2 \\ 2 \end{bmatrix}$$

$$W\vec{x} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \begin{bmatrix} 1 \\ -2 \\ 2 \end{bmatrix} = \begin{bmatrix} 1 - 4 + 6 \\ 4 - 10 + 12 \end{bmatrix} = \begin{bmatrix} 3 \\ 6 \end{bmatrix}$$

Example 2

$$W = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \quad \vec{x} = \begin{bmatrix} 1 \\ -2 \\ 2 \end{bmatrix} \quad \vec{b} = \begin{bmatrix} 7 \\ 10 \end{bmatrix}$$

$$W\vec{x} + \vec{b} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \begin{bmatrix} 1 \\ -2 \\ 2 \end{bmatrix} + \begin{bmatrix} 7 \\ 10 \end{bmatrix} = \begin{bmatrix} 3 \\ 6 \end{bmatrix} + \begin{bmatrix} 7 \\ 10 \end{bmatrix} = \begin{bmatrix} 10 \\ 16 \end{bmatrix}$$

Outline

- Review of linear algebra :
 - Vectors
 - Matrices
 - Matrix-vector multiplication
 - **Matrix-matrix multiplication**
 - Inner and outer products
 - Tensors
- PyTorch

Example 1

$$\begin{bmatrix} 1 & 1 & 1 \\ 2 & 3 & 0 \end{bmatrix} \begin{bmatrix} 1 & 2 & 1 & 1 \\ 2 & 0 & 1 & 1 \\ 3 & 2 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 6 & 4 & 3 & 3 \\ 8 & 4 & 5 & 5 \end{bmatrix}$$

(2×3) (3×4) $=$ (2×4)

$(\text{Row } 2) . (\text{col } 3) = (\text{entry } 2,3)$

Matrix-Matrix multiplication = batch of Matrix-Vector multiplication

$$\begin{bmatrix} 1 & 1 \\ 2 & -1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 5 \\ 1 \\ 3 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 \\ 2 & -1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 2 \\ 4 \end{bmatrix} = \begin{bmatrix} 6 \\ 0 \\ 4 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 \\ 2 & -1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 2 & 2 \\ 3 & 4 \end{bmatrix} = \begin{bmatrix} 5 & 6 \\ 1 & 0 \\ 3 & 4 \end{bmatrix}$$

Outline

- Review of linear algebra :
 - Vectors
 - Matrices
 - Matrix-vector multiplication
 - Matrix-matrix multiplication
 - Inner and outer products
 - Tensors
- PyTorch

Inner and outer products

$$\vec{x} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \quad \vec{y} = \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix}$$

$$\vec{x} \cdot \vec{y} = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix} = [8] \quad \text{Inner product}$$

(1 x 3) (3 x 1) = (1 x 1)

$$\vec{x} \otimes \vec{y} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \begin{bmatrix} 0 & 1 & 2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 2 \\ 0 & 2 & 4 \\ 0 & 3 & 6 \end{bmatrix} \quad \text{Outer product}$$

(3 x 1) (1 x 3) = (3 x 3)

Matrix-Matrix multiplication = sum of outer products

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 2 & 2 & 0 & 1 \\ 1 & 2 & 3 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 1 & 1 \\ 2 & 0 \\ 3 & 1 \end{bmatrix} = \begin{bmatrix} 6 & 3 \\ 5 & 3 \\ 0 & 2 \end{bmatrix}$$

This outer product

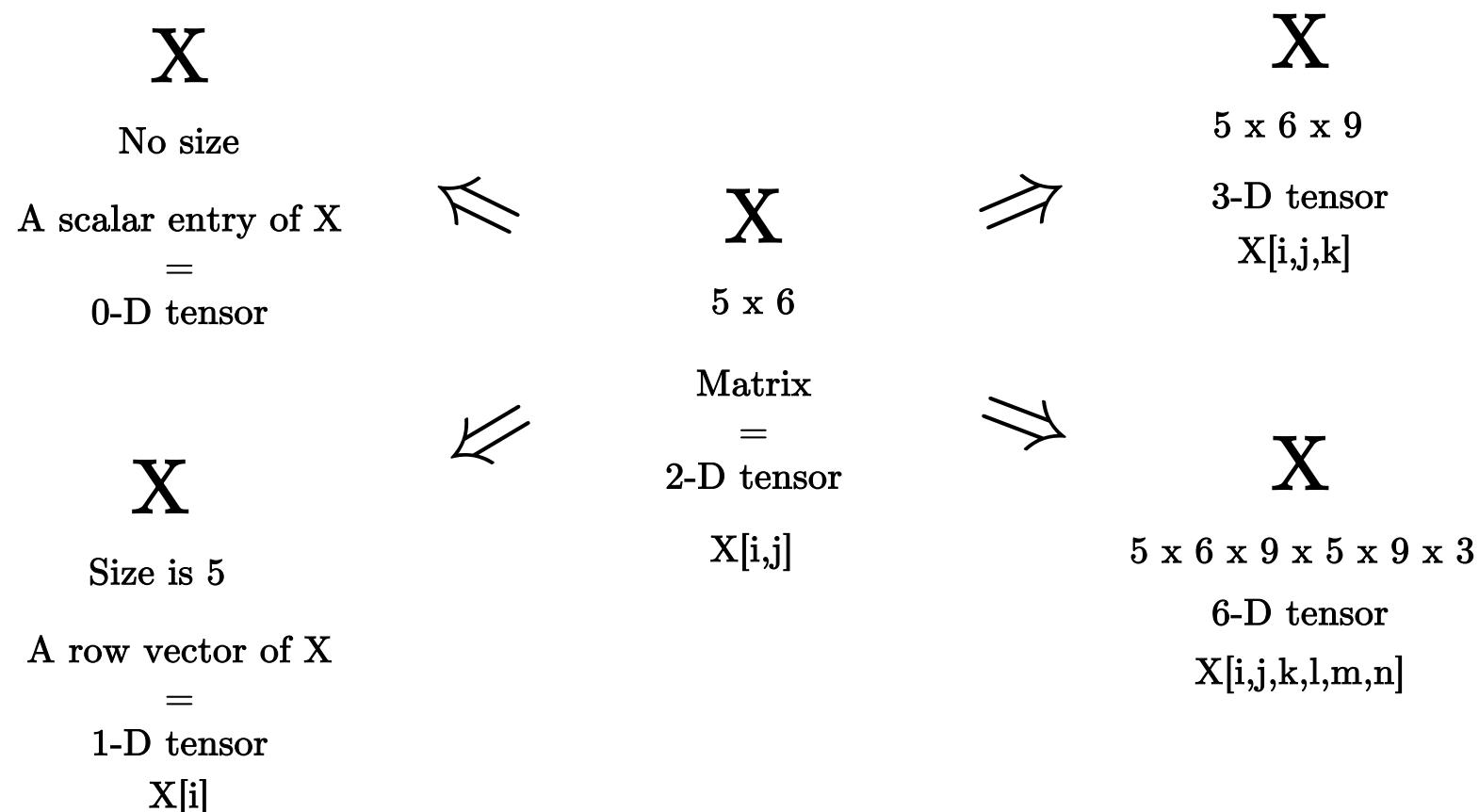
$$\overbrace{\begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} [0 \ 0] + \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix} [0 \ 1] + \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} [0 \ 0] + \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix} [1 \ 1] + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} [2 \ 0] + \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} [3 \ 1]}^{\text{This outer product}} = \begin{bmatrix} 6 & 3 \\ 5 & 3 \\ 0 & 2 \end{bmatrix}$$

Outline

- Review of linear algebra :
 - Vectors
 - Matrices
 - Matrix-vector multiplication
 - Matrix-matrix multiplication
 - Inner and outer products
 - Tensors
- PyTorch

Tensors

- Generalization of matrices to higher dimensions:

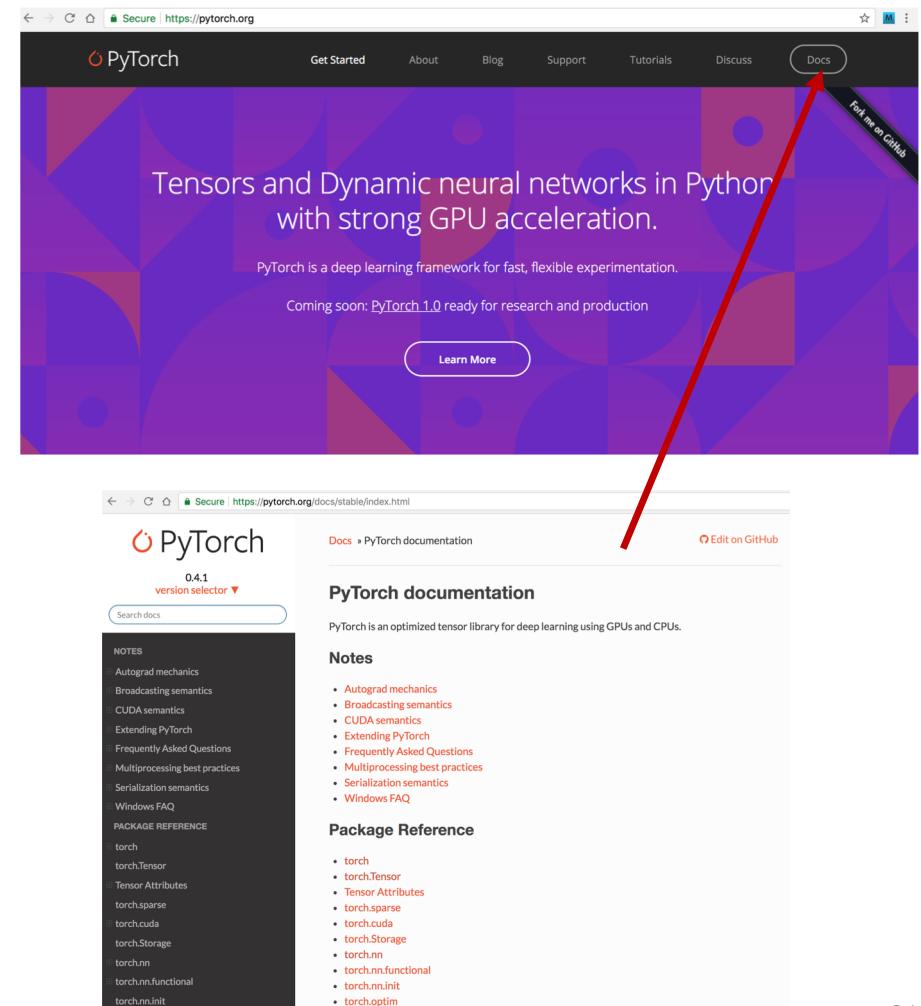


Outline

- Review of linear algebra :
 - Vectors
 - Matrices
 - Matrix-vector multiplication
 - Matrix-matrix multiplication
 - Inner and outer products
 - Tensors
- PyTorch

PyTorch

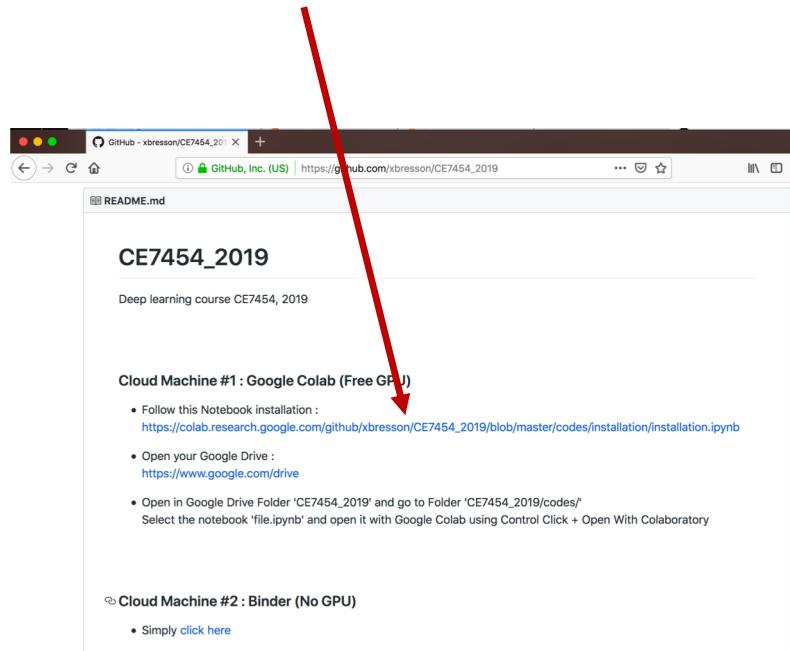
- PyTorch is an open source deep learning library in Python.
- Developed by FAIR (Facebook Artificial Intelligence Research).
- Original authors: Soumith Chintala, Adam Paszke, Sam Gross, Gregory Chanan.
- Initial release: October 2016
- PyTorch provides two high-level features:
 - Tensor computation (like numpy) with strong GPU acceleration
 - Deep Neural Networks built on an autodiff system



Cloud machine #1 (Free GPU)

- Follow these instructions :
 - Go to the GitHub folder of the course :
https://github.com/xbresson/CE7454_2019

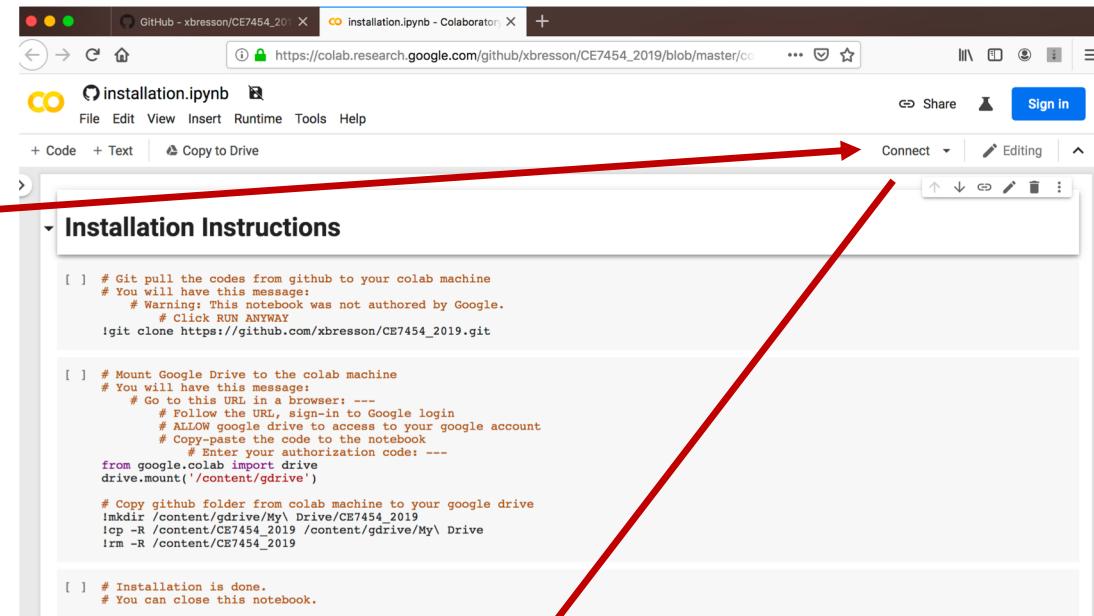
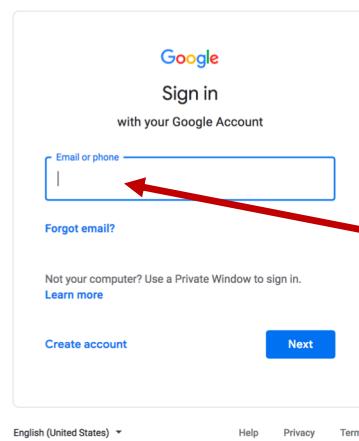
Click on this link.



The screenshot shows the Google Colaboratory interface. The top bar says 'Welcome To Colaboratory'. The main content area displays the 'Welcome to Colaboratory!' page, which includes a video thumbnail for 'Intro to Google Colab' featuring a smiling man.

Cloud machine #1 (Free GPU)

- Click on CONNECT.
- It will ask you to sign-in with your Gmail account.



A screenshot of a Google Colab notebook titled "installation.ipynb". The notebook contains Python code for cloning a GitHub repository and mounting Google Drive. A red arrow points from the "CONNECT" button in the list above to the "Connect" button in the top right corner of the Colab interface. Another red arrow points from the "Email or phone" input field in the sign-in box to the "SIGN IN" button in the modal dialog.

```
[ ] # Git pull the codes from github to your colab machine
# You will have this message:
# Warning: This notebook was not authored by Google.
# Click RUN ANYWAY
!git clone https://github.com/xbresson/CE7454_2019.git

[ ] # Mount Google Drive to the colab machine
# You will have this message:
# Go to this URL in a browser: ---
# Follow the URL, sign-in to Google login
# ALLOW google drive to access to your google account
# Copy-paste the code to the notebook
# Enter your authorization code: ---
from google.colab import drive
drive.mount('/content/gdrive')

# Copy github folder from colab machine to your google drive
!mkdir /content/gdrive/My\ Drive/CE7454_2019
!cp -R /content/CE7454_2019 /content/gdrive/My\ Drive
!rm -R /content/CE7454_2019

[ ] # Installation is done.
# You can close this notebook.
```

Google sign-in required

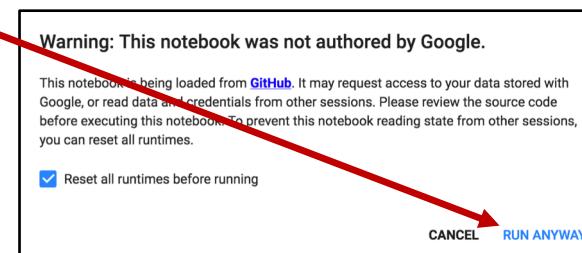
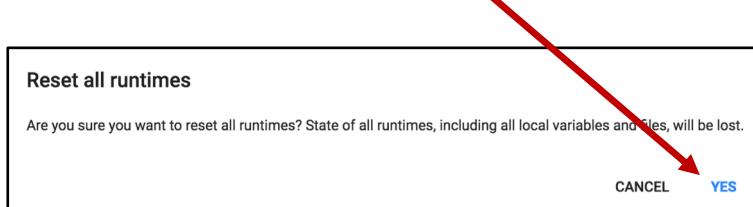
You must be logged in to a Google account to continue.

CANCEL SIGN IN

Cloud machine #1 (Free GPU)

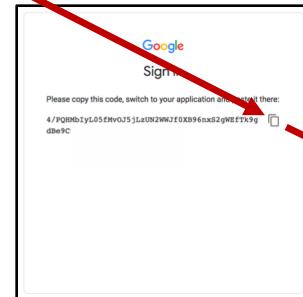
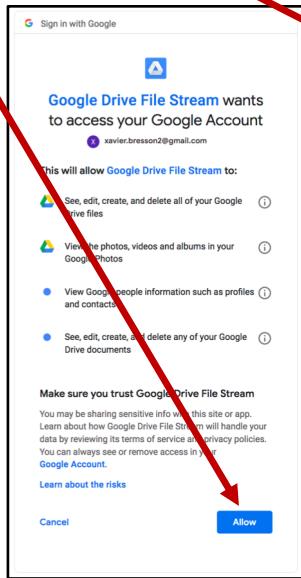
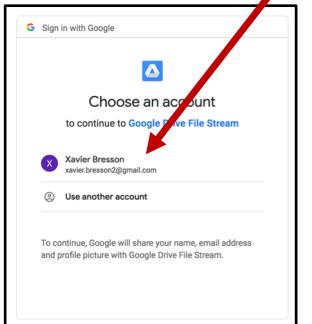
- Click on **CONNECT** again to start the Google Cloud machine.
- Run the first cell to clone the codes from GitHub to the Google Cloud machine.
 - It will give a warning, click on **RUN ANYWAY**.
 - Answer **YES** to the next question **RESET ALL RUNTIMES**.

```
[ ] # Git pull the codes from github to your colab machine  
# You will have this message:  
# Warning: This notebook was not authored by Google.  
!git clone https://github.com/xbresson/CE7454_2019.git  
  
[ ] # Mount Google Drive to the colab machine  
# You will have this message:  
# Go to this URL in a browser: ---  
# Follow the URL, sign-in to Google login  
# ALLOW google drive to access to your google account  
# Copy-paste the code to the notebook  
# Enter your authorization code: ---  
from google.colab import drive  
drive.mount('/content/gdrive')  
  
# Copy github folder from colab machine to your google drive  
!mkdir /content/gdrive/My\ Drive/CE7454_2019  
!cp -R /content/CE7454_2019 /content/gdrive/My\ Drive  
!rm -R /content/CE7454_2019  
  
[ ] # Installation is done.  
# You can close this notebook.
```



Cloud machine #1 (Free GPU)

- Run the second cell to mount your Google Drive to the Google Cloud machine (all your codes will be saved in Google Drive).
 - Click on the provided URL.
 - Select your Gmail account.
 - ALLOW Google Drive File Stream.
 - Copy-paste the code to the notebook (Enter your authorization code) and press Return.



The screenshot shows a Jupyter Notebook cell with the following content:

```
[1] # Git pull the codes from github to your colab machine
# You will have this message:
# Warning: This notebook was not authored by Google.
# Click RUN ANYWAY
!git clone https://github.com/xbresson/CE7454_2019.git
```

Cloning into 'CE7454_2019'...
remote: Enumerating objects: 223, done.
remote: Counting objects: 100% (223/223), done.
remote: Compressing objects: 100% (135/135), done.
remote: Total 223 (delta 100), reused 191 (delta 72), pack-reused 0
Receiving objects: 100% (223/223), 2.63 MiB | 6.68 MiB/s, done.
Resolving deltas: 100% (100/100), done.

```
# Mount Google Drive to the colab machine
# You will have this message:
# Go to this URL in a browser: ---
# Follow the URL, sign-in to Google login
# ALLOW google drive to access to your google account
# Copy-paste the code to the notebook
# Enter your authorization code: ---
```

```
from google.colab import drive
drive.mount('/content/gdrive')
```

```
# Copy github folder from colab machine to your google drive
!mkdir /content/gdrive/My\ Drive/CE7454_2019
!cp -R /content/CE7454_2019 /content/gdrive/My\ Drive
!rm -R /content/CE7454_2019
```

... Go to this URL in a browser: https://accounts.google.com/o/oauth2/auth?client_id=947318989803-6bn6qk8g

Enter your authorization code:

Cloud machine #1 (Free GPU)

- Open your Google Drive :
<https://www.google.com/drive>
- Go folder CS5242_2021_codes

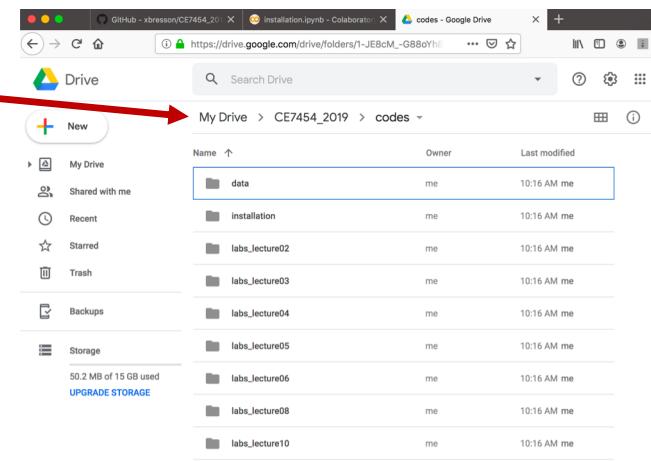
The image contains three screenshots of the Google Drive web interface:

- Screenshot 1 (Top):** Shows the main Google Drive landing page with a wooden desk background. It features a large blue "Go to Google Drive" button in the center.
- Screenshot 2 (Middle):** Shows a list view of files in a folder named "CE7454_2019". The list includes ".git", "codes", ".gitignore", "environment.yml", "LICENSE", and "README.md". A red arrow points from the text "Click here for List View" to the list icon in the top right corner of this screenshot.
- Screenshot 3 (Bottom):** Shows a list view of files in the same folder. The list includes ".git", "codes", ".gitignore", "environment.yml", "LICENSE", and "README.md". A red arrow points from the text "Click here for List View" to the list icon in the top right corner of this screenshot.

Click here for List View

Cloud machine #1 (Free GPU)

- Go folder CS5242_2021_codes/codes
- Let us open the notebook `pytorch_introduction.ipynb` in folder codes/labs_lecture02.
 - Select the notebook and open it using Control Click + Open With Colaboratory



A composite screenshot illustrating the process of opening a Jupyter notebook in Google Colaboratory. On the left, a screenshot of Google Drive shows the 'pytorch_introduction.ipynb' file selected in the 'labs_lecture02' folder. A red arrow points from the text 'Select the notebook and open it using Control Click + Open With Colaboratory' to the 'Colaboratory' option in the context menu. On the right, a screenshot of the Colab interface shows the notebook titled 'Lab 01: Introduction to PyTorch' running. The code cell contains the following Python code:

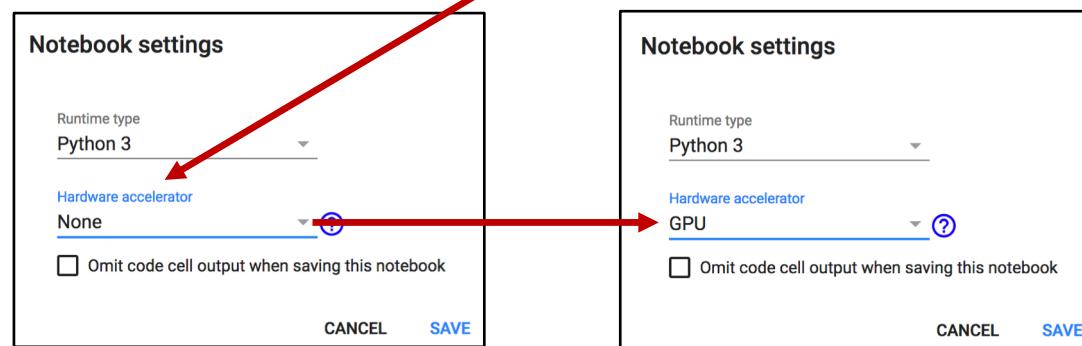
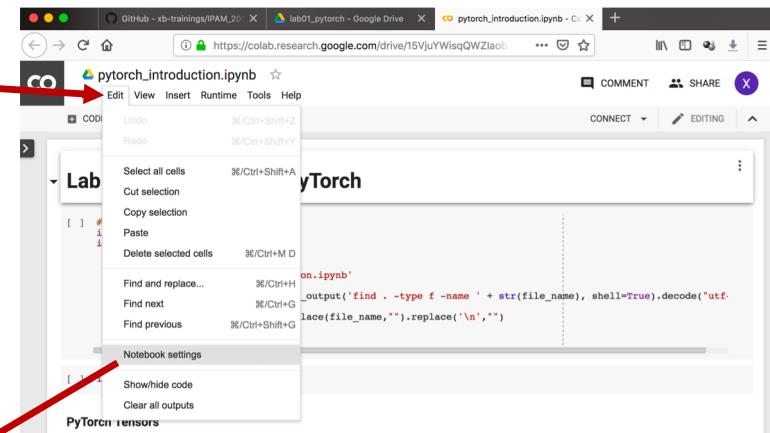
```
# For Google Colaboratory
import sys, os
if 'google.colab' in sys.modules:
    from google.colab import drive
    drive.mount('/content/drive')
    file_name = 'pytorch_introduction.ipynb'
    import subprocess
    path_to_file = subprocess.check_output('find . -type f -name ' + str(file_name), shell=True).decode("utf-8")
    path_to_file_to_file = path_to_file.replace(file_name,'').replace('\n','')
    os.chdir(path_to_file)
    ipynb
```

The code cell below imports torch and creates a tensor: `[] x=torch.Tensor([5.3, 2.1, -3.1]) print(x)` resulting in `tensor([5.3000, 2.1000, -3.1000])`.

Cloud machine #1 (Free GPU)

- GPU accelerator :

- Select **Edit** in the menu and **Notebook settings**.
- Select **GPU** in **Hardware accelerator**.



Cloud machine #2 (No GPU)

- Start your **cloud machine** :
 - Type this address in your favorite web browser :
https://mybinder.org/v2/gh/xbresson/CS242_2021/master
 - Binder creates **mini-instances** on Amazon Cloud.
 - Binder provides at least 12 hours of session time per user session.
 - Binder will automatically **shut down** user sessions that have more than 10 minutes of **inactivity** (if you leave your window open, this will be counted as “**activity**”).

A screenshot of a web browser displaying the Binder interface. The address bar shows the URL https://hub.mybinder.org/user/xb-trainings-ipam_2019-yrvgnqq4/tree. The page title is "jupyter". Below the title, there are tabs for "Files", "Running", and "Clusters", with "Files" selected. A message says "Select items to perform actions on them." Below this is a file tree:

Name	Last Modified	File size
codes	26 minutes ago	294 B
environment.yml	26 minutes ago	294 B
LICENSE	26 minutes ago	1.07 kB
README.md	26 minutes ago	671 B

Your local machine

- Install Python and run the notebooks **on your machine** (it will use the GPU if you have it) :
 - Follow these instructions (easy steps) :

Running Python notebooks on your computer

Installation

```
# Conda installation for Linux and OSX
# Open a Terminal
curl -o ~/miniconda.sh -0 https://repo.continuum.io/miniconda/Miniconda3-latest-Linux-x86_64.sh # Linux
curl -o ~/miniconda.sh -0 https://repo.continuum.io/miniconda/Miniconda3-latest-MacOSX-x86_64.sh # OSX
chmod +x ~/miniconda.sh
./miniconda.sh
source ~/.bashrc

# Conda installation for Windows
# Download and install : https://repo.anaconda.com/miniconda/Miniconda3-latest-Windows-x86_64.exe
# Open new Application "Anaconda Prompt Terminal"

# Clone GitHub repo
conda install git
git clone https://github.com/xbresson/CS5242_2021.git
cd CS5242_2021

# Install python libraries
conda env create -f environment.yml
source activate deeplearn_course

# Run the notebooks
jupyter notebook
```

Lab 01

- Introduction to PyTorch

The screenshot shows a Jupyter Notebook interface with the title "Lab 01: Introduction to PyTorch". The notebook has four code cells:

- In [1]:** `import torch`
- PyTorch Tensors**
Construct a vector of 3 elements
In [2]: `x=torch.Tensor([5.3 , 2.1 , -3.1])
print(x)`
Output: `tensor([5.3000, 2.1000, -3.1000])`
- Construct a 2 x 2 matrix**
In [3]: `A=torch.Tensor([[5.3,2.1] , [0.2,2.1]])
print(A)`
Output: `tensor([[5.3000, 2.1000],
[0.2000, 2.1000]])`
- Construct a random 10 x 2 matrix**
In [4]: `A=torch.rand(10,2)
print(A)`

Lab 02

- Manipulating PyTorch tensors – basic linear algebra

Lab 03

- Manipulating PyTorch tensors – reshaping and slicing

jupyter pytorch_tensor_part2_exercise Last Checkpoint: a few seconds ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

In []: `import torch
import utils`

Make a 10×2 matrix random matrix A. Then store its third row (index = 2) in to a vector v. Then store the first 5 rows (index 0 to index 4) into a submatrix B. The important information is that B has a total of five rows. Print A, v and B.

In []: `# write your code here`

Extract entry (0,0) of the matrix A and store it into a PYTHON NUMBER x

In []: `# write your code here`

Let's download 60,000 gray scale pictures as well as their label. Each picture is 28 by 28 pixels.

In []: `from utils import check_mnist_dataset_exists
data_path=check_mnist_dataset_exists()

data=torch.load(data_path+'mnist/train_data.pt')
label=torch.load(data_path+'mnist/train_label.pt')`

More tutorials

- Learning PyTorch with Examples

The screenshot shows a web browser displaying the PyTorch tutorial page at https://pytorch.org/tutorials/beginner/pytorch_with_examples.html. The page is titled "Learning PyTorch with Examples". The left sidebar contains a navigation menu with sections for "BEGINNER TUTORIALS" (Deep Learning with PyTorch: A 60 Minute Blitz, PyTorch for former Torch users, Learning PyTorch with Examples), "INTERMEDIATE TUTORIALS" (Transfer Learning tutorial, Data Loading and Processing Tutorial, Deep Learning for NLP with Pytorch), and "nn module". The main content area starts with a note: "This tutorial introduces the fundamental concepts of PyTorch through self-contained examples. At its core, PyTorch provides two main features: An n-dimensional Tensor, similar to numpy but can run on GPUs; Automatic differentiation for building and training neural networks. We will use a fully-connected ReLU network as our running example. The network will have a single hidden layer, and will be trained with gradient descent to fit random data by minimizing the Euclidean distance between the network output and the true output." Below this is a "Note" section stating "You can browse the individual examples at the [end of this page](#)". A "Table of Contents" sidebar on the right lists: Tensors (Warm-up: numpy, PyTorch: Tensors), Autograd (PyTorch: Tensors and autograd, PyTorch: Defining new autograd functions, TensorFlow: Static Graphs), and nn module.

https://pytorch.org/tutorials/beginner/pytorch_with_examples.html



Questions?