ECE C147/C247, Winter 2022
Neural Networks & Deep Learning
UCLA ECE

Setting up Google Colab
Prof. J.C. Kao
TAs: Y. Li, P. Lu, T. Monsoor, T. Wang

In this class, we will use Python, and in particular, we will use Jupyter notebooks for assignments. You will gain exposure to this in Homework #1.

This guide will show you how to load Jupyter Notebooks onto Google Colab, a free resource for running Python notebooks. Each time you open a Colab notebook, you get the same version of Python (hosted by Google) that is preloaded with many useful libraries. This way you don't need to worry about managing your own virtual environments, since Google does it for you!

Uploading Notebooks to Colab. Assignments will include files with extension `.ipynb`, which is the file format for Python notebooks. To upload these notebooks onto Google Colab, simply go to *File* > *Upload Notebook* and upload this file. This may only be the first step, though, since the notebook may depend on other resources. See Uploading Resources.

Uploading Resources. Your python notebooks will depend on resources external to the `.ipynb` file, such as other Python `.py` files and a `requirements.txt` listing the Python packages required. There are two ways in which you can install all the packages mentioned in `requirements.txt`.

1. You can upload the `requirements.txt` file as a temporary file by just uploading it to Google Colab during runtime. For uploading it during runtime, all you have to do is select the ”Upload files” option on left. (Shown below)

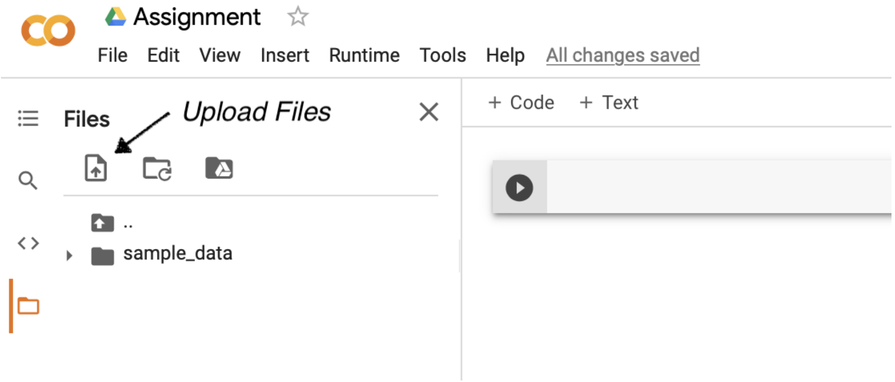

Figure 1: Caption

The thing to note over here is that, this storage is temporary. So if you close your colab session and you would again like to run `requirements.txt` file, you would have to upload it again. For a more long-term solution, you can refer to the second option.

2. The second option is to upload `requirements.txt` to Google drive. Once you have uploaded your file to to Google drive you can mount your google drive to google colab and access files in google drive.

The code below shows how to mount your Google Drive on your runtime using an authorization code and then run `requirements.txt` file. (Specify the file path appropriately).

```
from google.colab import drive
drive.mount('/content/drive')
!pip install -r '/content/drive/My Drive/requirements.txt'
```

Executing Bash and Installing Dependencies. One attractive aspect of Colab is the ability to run bash commands directly in a Colab cell. For example, installing PyTorch is as simple as running a cell with the entry

```
!pip install torch
```

This can be a really useful feature for projects, though most of the homework assignments won't really use this feature much. The main way you will be using this is by installing the assignment dependencies located in the `requirements.txt` file. Once you've uploaded the `requirements.txt` as a resource, you can create and run a cell containing

```
!pip install -r requirements.txt
```

to install the dependencies. **Note: You will have to do this everytime you restart your runtime.**

Comments about Colab. Colab is a great resource, and it can even give you access to GPUs to speed up your notebooks. But it has some drawbacks. First of all, Colab will stop running if you aren't active on the webpage. A workaround for that is to run some Javascript in your browser's console to constantly click on page elements to keep the page active (see this Medium post or this Stack Overflow thread for explanations on how to do this). A worse problem is that even if you do this, Colab will eventually timeout and block you from using their resources for a while (they say after 12 hours of use). Be wary of this possibility if you are planning on using Colab; if you plan on using Colab for your final project, be sure to save checkpoints of your models during training so that your work isn't lost during a timeout (which has happened to this author far too many times...).