Science & Technology Facilities Council
Rutherford Appleton Laboratory

Machine Learning Tutorial Overview

Emmanuel Olaiya

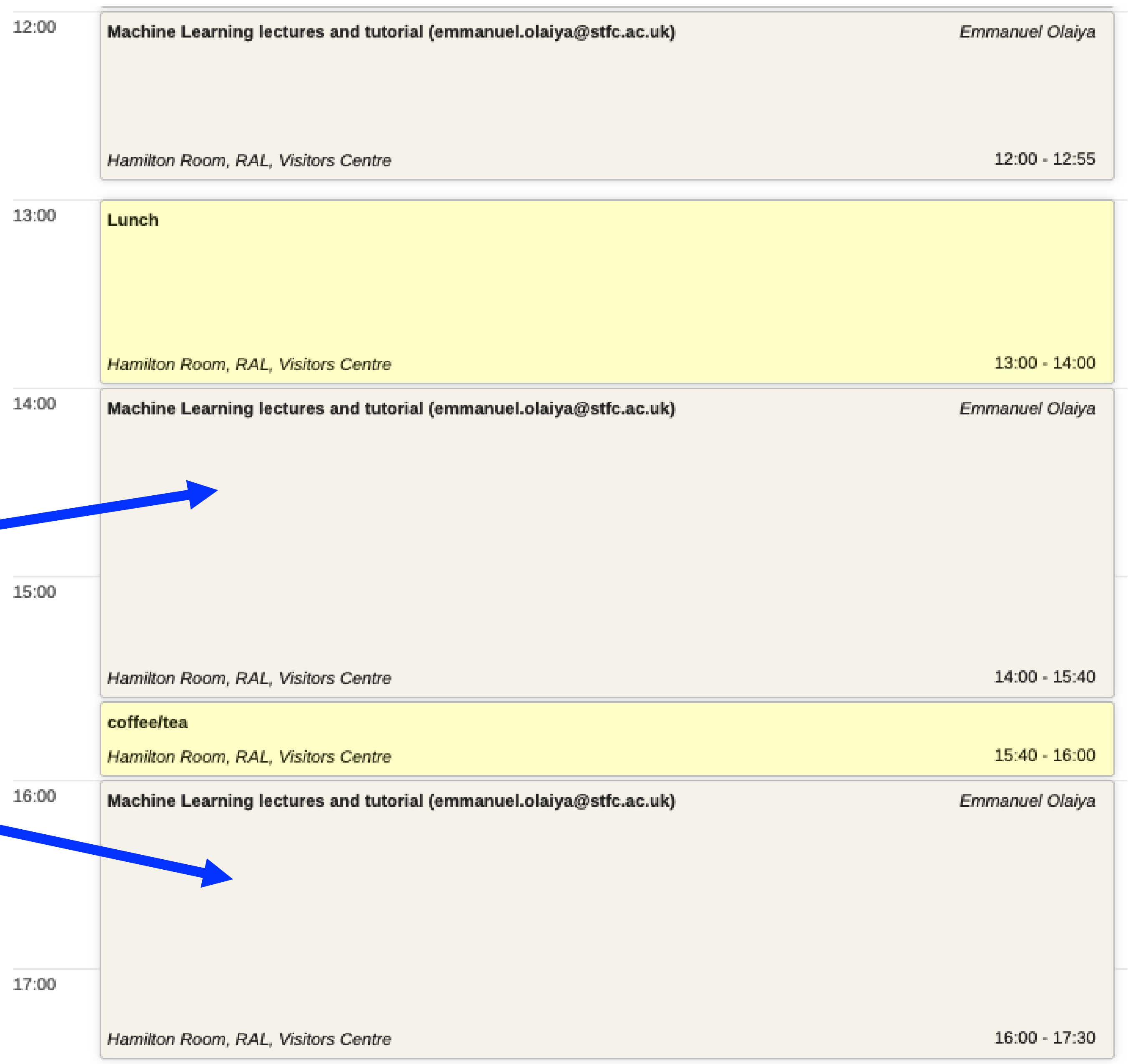
Rutherford Appleton Laboratory

16th June 2024

Agenda

Today, Moday 3rd June, this afternoon

- Tutorial walkthrough
- 1400 - 1540
- We will have breaks
- 1600 - 1730



12:00

Machine Learning lectures and tutorial (emmanuel.olaiya@stfc.ac.uk) *Emmanuel Olaiya*

Hamilton Room, RAL, Visitors Centre 12:00 - 12:55

13:00

Lunch

Hamilton Room, RAL, Visitors Centre 13:00 - 14:00

14:00

Machine Learning lectures and tutorial (emmanuel.olaiya@stfc.ac.uk) *Emmanuel Olaiya*

15:00

Hamilton Room, RAL, Visitors Centre 14:00 - 15:40

coffee/tea

Hamilton Room, RAL, Visitors Centre 15:40 - 16:00

16:00

Machine Learning lectures and tutorial (emmanuel.olaiya@stfc.ac.uk) *Emmanuel Olaiya*

17:00

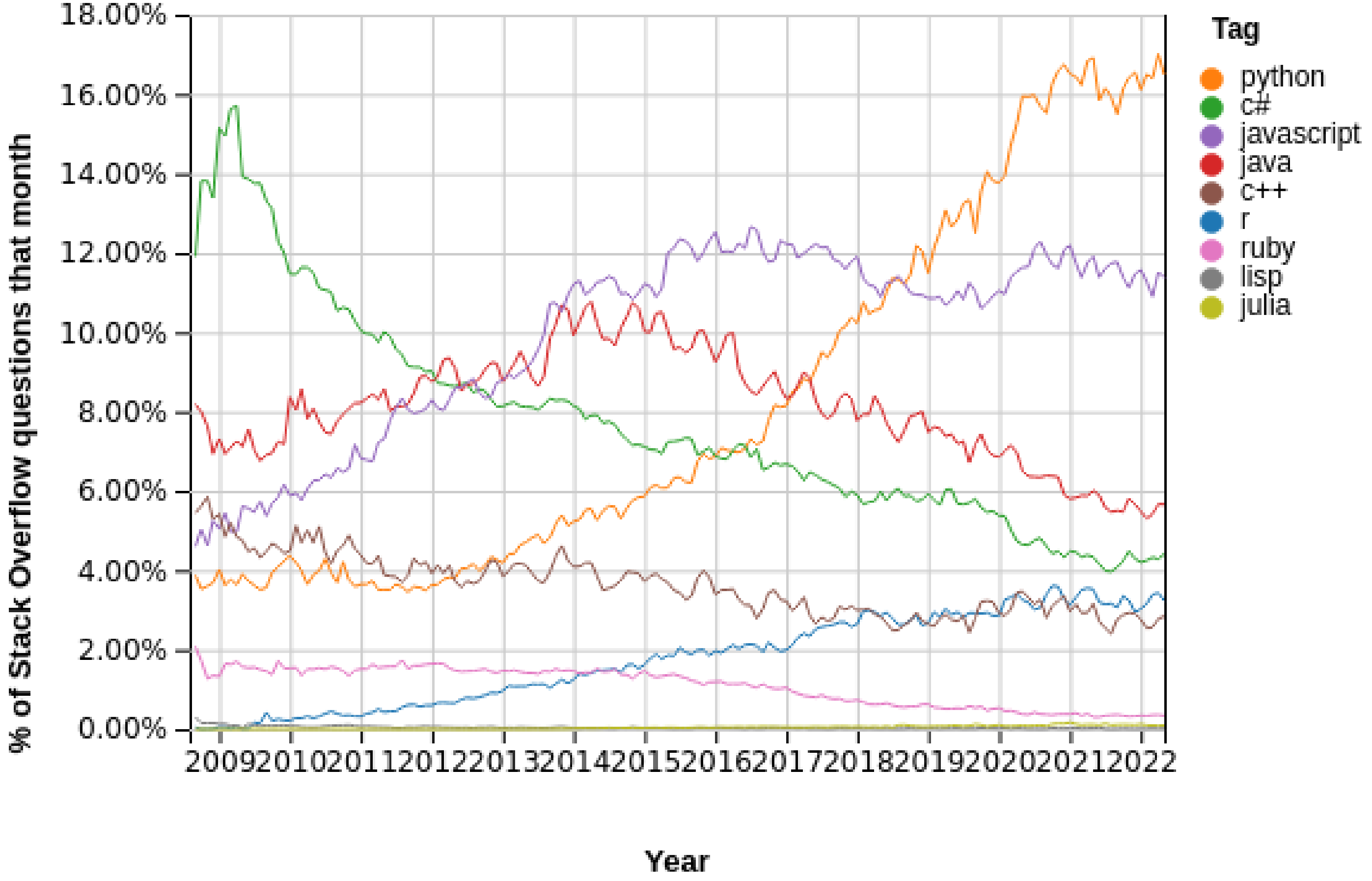
Hamilton Room, RAL, Visitors Centre 16:00 - 17:30

Content

- What is the goal of the tutorial
    - Continue the learning. Learn by doing!
    - Give you some hands on experience
    - Introduce you to some simple tools that have a wide application
- Again this is an introduction. So the examples will be straightforward. Anyone who has done any machine learning before may find it easy. This workbook is for beginners!
- The tutorial will be a walkthrough
    - It will be interactive. You will be able to execute code. Modify it if you want
    - There will be some exercises for you to practice implementing machine learning code

Contents

- What programming languages are used for machine learning
  - Python, R, Java, Julia, LISP, C++, ….. , lots!
- Python is a language that continues to grow in popularity. Python is the language we will use



Contents

- So what are the machine learning toolkits available
    - Again lots: Tensorflow, Pytorch, Sklearn, Amazon Machine Learning (AML), Shogun, ….
    - We will be using Tensorflow, a platform owned by Google, it is well supported, easy to use and has a wide range of features
        - GPU support is seamless (for Nvidia cards!)
            - Good at detecting if you have GPUs and then just using them without any need for configuring

Jupyter Notebooks

- We will use Jupyter Notebooks for this tutorial
- What are they?
- They are a web-based Interactive Develeopment Environment (IDE), from which you can program and run code

How do we run the notebooks

- There are many ways to run the notebooks. I am going to suggest 4

  If you have a preferable method then that is fine

  My suggestions are:

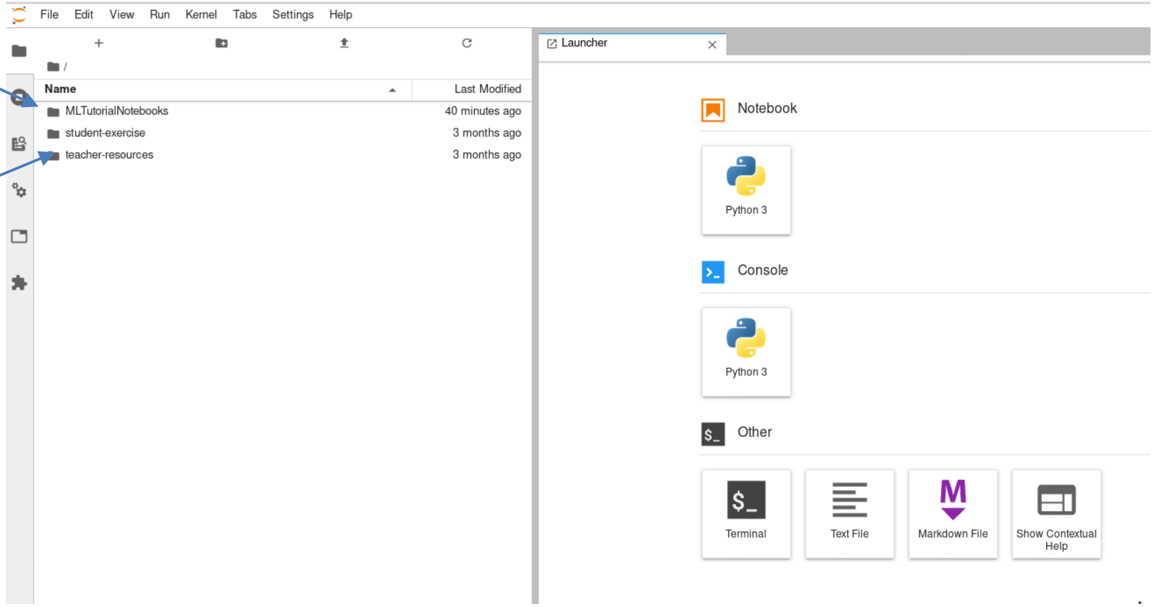
1. Using the RAL computing cluster
2. Using mybinder.org
3. Using Google Colab
4. Running on your own computer

Using the RAL Cluster

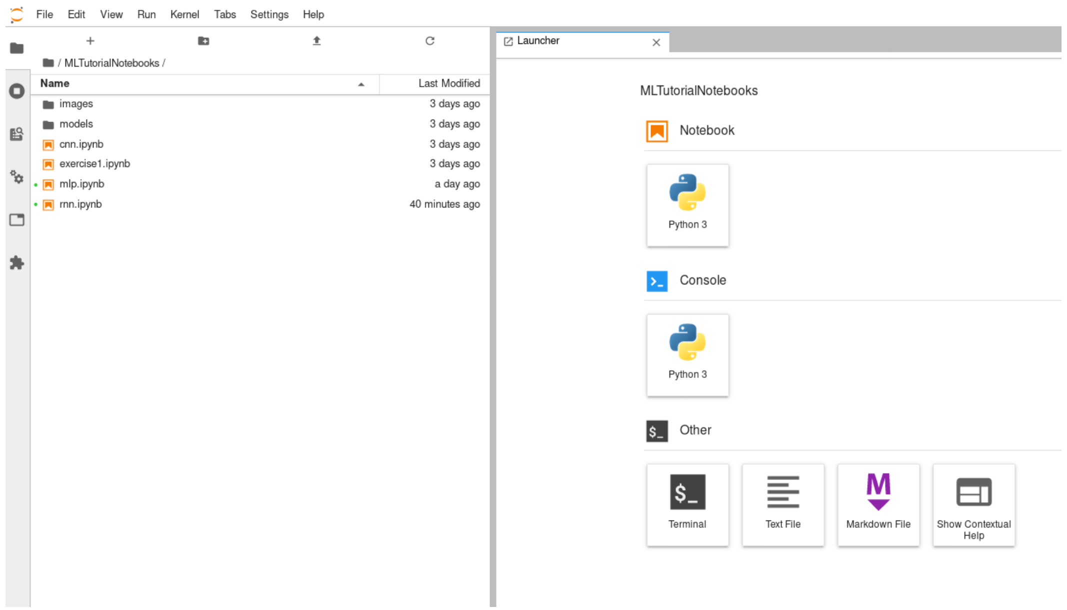
- If you want to use the RAL cluster I will provide you with a username and a password
- Go to the page https://monty-backup.stfc.ac.uk Log in with your username and password. You should see the following:

Click on MLTutorialNotebooks

The student-exercise and teacher-exercise tabs won't be there



Name	Last Modified
MLTutorialNotebooks	40 minutes ago
student-exercise	3 months ago
teacher-resources	3 months ago

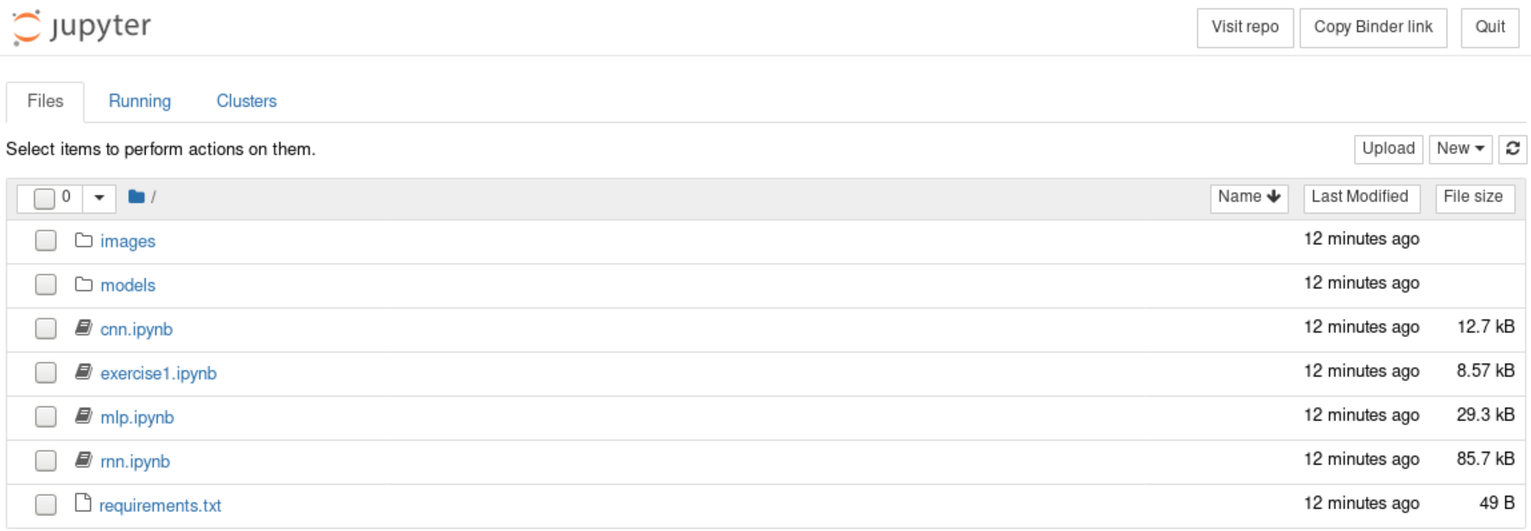


Name	Last Modified
images	3 days ago
models	3 days ago
cnn.ipynb	3 days ago
exercise1.ipynb	3 days ago
mlp.ipynb	a day ago
rnn.ipynb	40 minutes ago

Click on one of the .ipynb files to run the notebook

Using mybinder.org

- Everyone can use this option
- Just click on this link:
  - https://mybinder.org/v2/gh/olaiya/MLTutorialNotebooks.git/HEAD
- After a while you should be able to see the notebook



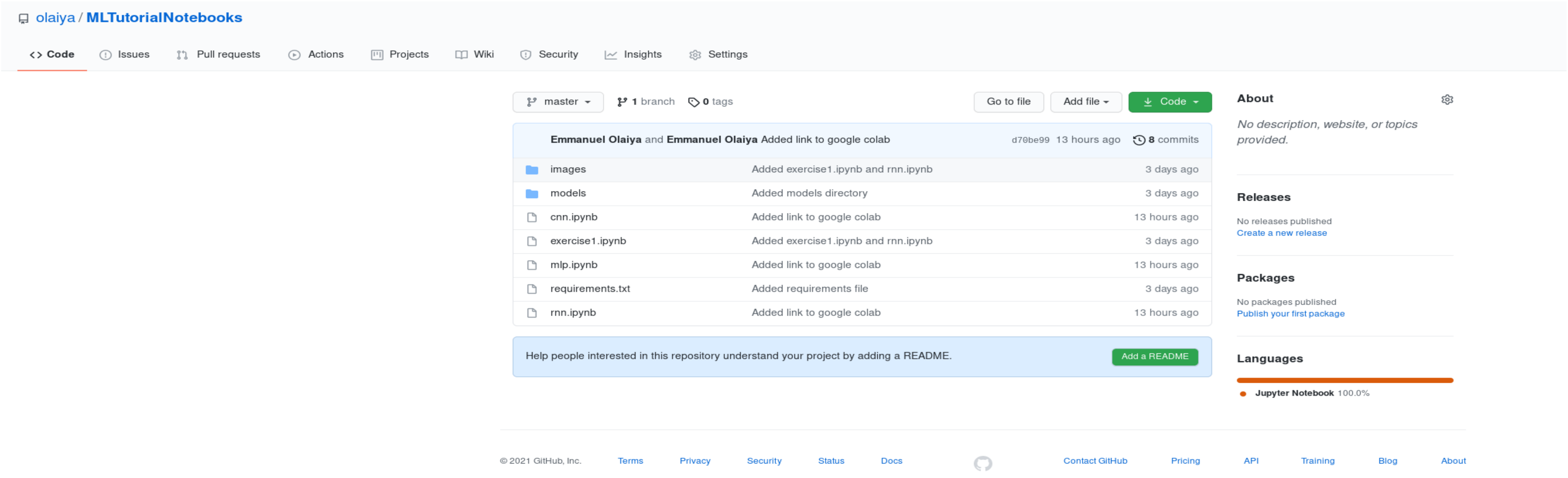
Select items to perform actions on them.

Name	Last Modified	File size
images	12 minutes ago	
models	12 minutes ago	
cnn.ipynb	12 minutes ago	12.7 kB
exercise1.ipynb	12 minutes ago	8.57 kB
mlp.ipynb	12 minutes ago	29.3 kB
rnn.ipynb	12 minutes ago	85.7 kB
requirements.txt	12 minutes ago	49 B

If you are using mybinder, connect to the above link 5 mins before the start of the tutorial.
Sometimes it can take a few minutes to fire up the container that hosts the notebook

Using Google Colab

- People with a Google account can use this option
- Make sure you are logged into your Google account
- Go to the workbook files on github:
  - https://github.com/olaiya/MLTutorialNotebooks



Emmanuel Olaiya and Emmanuel Olaiya Added link to google colab d70be99 13 hours ago 8 commits

images	Added exercise1.ipynb and rnn.ipynb	3 days ago
models	Added models directory	3 days ago
cnn.ipynb	Added link to google colab	13 hours ago
exercise1.ipynb	Added exercise1.ipynb and rnn.ipynb	3 days ago
mlp.ipynb	Added link to google colab	13 hours ago
requirements.txt	Added requirements file	3 days ago
rnn.ipynb	Added link to google colab	13 hours ago

Help people interested in this repository understand your project by adding a README.

No description, website, or topics provided.

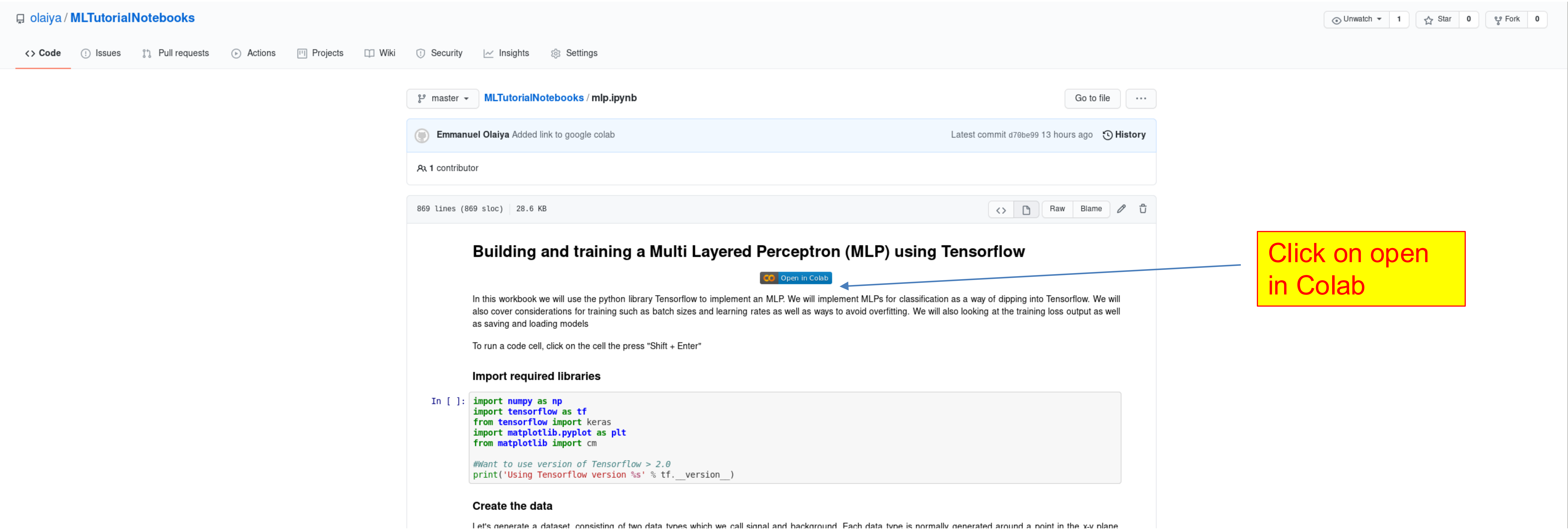
No releases published
Create a new release

No packages published
Publish your first package

● Jupyter Notebook 100.0%

Using Google Colab

- When we work on a specific workbook, say mlp.ipynb, click on it



MLTutorialNotebooks / mlp.ipynb

Emmanuel Olaiya Added link to google colab

Latest commit d70be99 13 hours ago History

1 contributor

869 lines (869 sloc) 28.6 KB

Building and training a Multi Layered Perceptron (MLP) using Tensorflow

In this workbook we will use the python library Tensorflow to implement an MLP. We will implement MLPs for classification as a way of dipping into Tensorflow. We will also cover considerations for training such as batch sizes and learning rates as well as ways to avoid overfitting. We will also looking at the training loss output as well as saving and loading models

To run a code cell, click on the cell the press "Shift + Enter"

Import required libraries

In [ ]:

```
import numpy as np
import tensorflow as tf
from tensorflow import keras
import matplotlib.pyplot as plt
from matplotlib import cm

#Want to use version of Tensorflow > 2.0
print('Using Tensorflow version %s' % tf.__version__)
```

Create the data

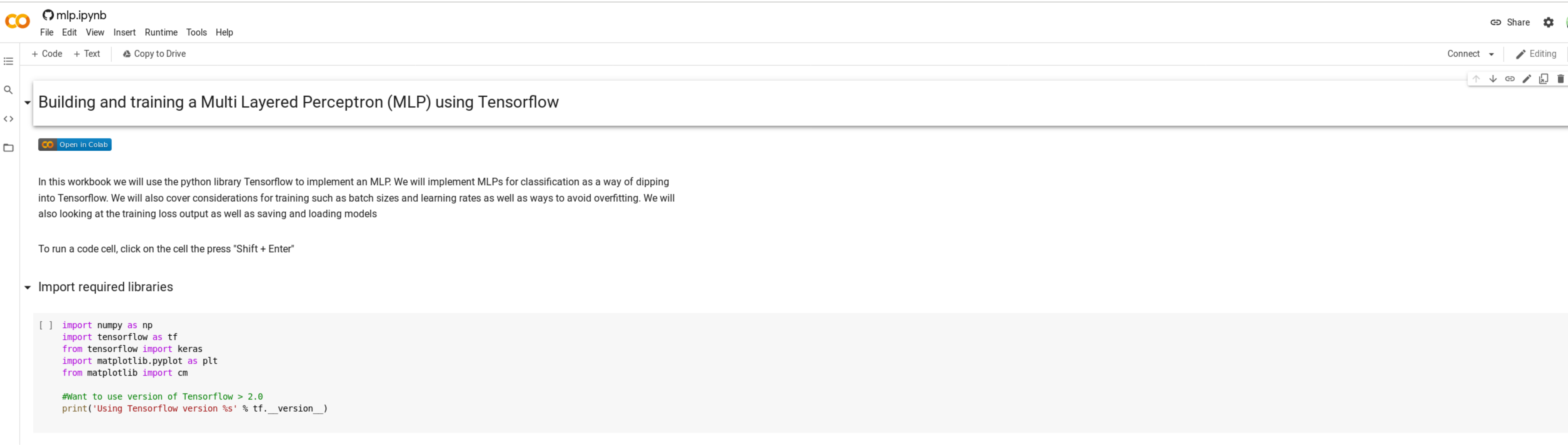
Let's generate a dataset, consisting of two data types which we call signal and background. Each data type is normally generated around a point in the x-y plane

Click on open in Colab

- If clicking on icon doesn’t work, copy and paste link in your browser

Using Google Colab

- The workbook should load


mlp.ipynb
File Edit View Insert Runtime Tools Help
Share
+ Code + Text Copy to Drive
Connect Editing
Open in Colab

Building and training a Multi Layered Perceptron (MLP) using Tensorflow

In this workbook we will use the python library Tensorflow to implement an MLP. We will implement MLPs for classification as a way of dipping into Tensorflow. We will also cover considerations for training such as batch sizes and learning rates as well as ways to avoid overfitting. We will also looking at the training loss output as well as saving and loading models

To run a code cell, click on the cell the press "Shift + Enter"

Import required libraries

```
[ ] import numpy as np
    import tensorflow as tf
    from tensorflow import keras
    import matplotlib.pyplot as plt
    from matplotlib import cm

    #Want to use version of Tensorflow > 2.0
    print('Using Tensorflow version %s' % tf.__version__)
```

Using Your Own Computer

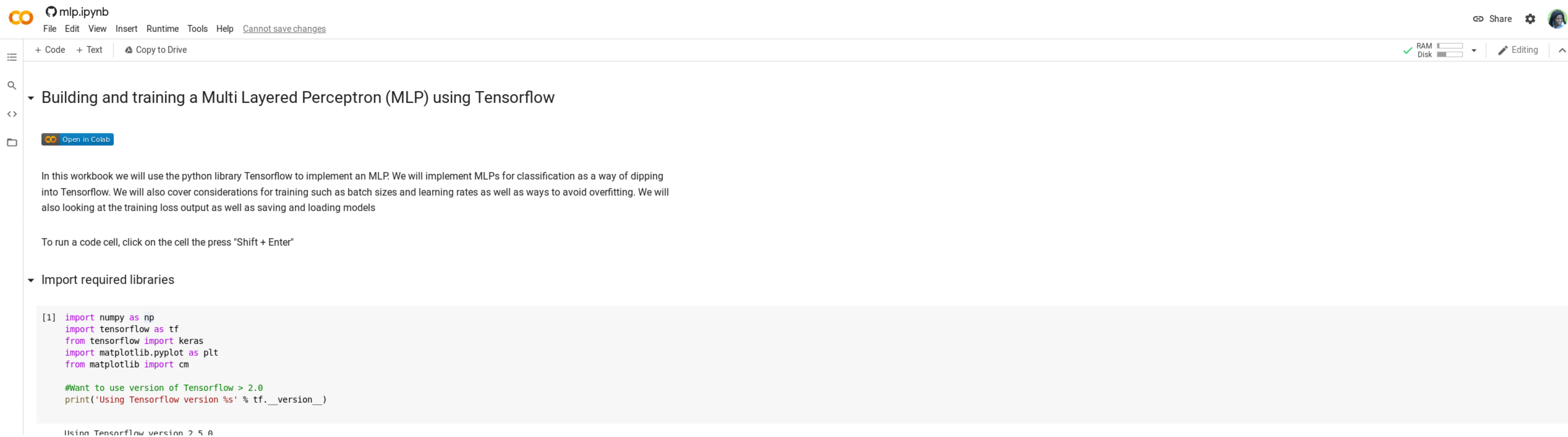
- If you want to run the workbook on your own computer that is fine
- Workbook is intentionally lightweight for this purpose
- Many setups you can use, install all the software, use a container such as Docker, use Conda ………
- What you need is git, python3 and the following libraries, tensorflow (>2.0.0), numpy, pandas, sklearn, matplotlib, jupyter
  - You don’t need a GPU
- To pull the workbook onto you computer run:
  - git clone https://github.com/olaiya/MLTutorialNotebooks.git
  - See the requirements.txt file for the python modules required
    - pip install –r requirements.txt
    - conda install –file requirements.txt

  Or download the workbook directly from the webpage: https://github.com/olaiya/MLTutorialNotebooks

- Run jupyter notebook

Running A Workbook

- Workbooks are a collection of cells. The cells are either code cells or markdown cells (adding text or images)
- To run a piece of code in a cell, select the cell and press Shift+Enter


mlp.ipynb
File Edit View Insert Runtime Tools Help Cannot save changes
Share
+ Code + Text Copy to Drive
RAM Disk
Editing
Open in Colab

Building and training a Multi Layered Perceptron (MLP) using Tensorflow

In this workbook we will use the python library Tensorflow to implement an MLP. We will implement MLPs for classification as a way of dipping into Tensorflow. We will also cover considerations for training such as batch sizes and learning rates as well as ways to avoid overfitting. We will also looking at the training loss output as well as saving and loading models

To run a code cell, click on the cell the press "Shift + Enter"

Import required libraries

```
[1]  import numpy as np
     import tensorflow as tf
     from tensorflow import keras
     import matplotlib.pyplot as plt
     from matplotlib import cm

     #Want to use version of Tensorflow > 2.0
     print('Using Tensorflow version %s' % tf.__version__)

     Using Tensorflow version 2.5.0
```

- To add a cell click Insert->Code/Text cell
  - Some instances Insert->Cell Above/Below
    - Then select Cell->Cell type (to change between code and markdown)

Today's Workbook Tutorial

- Today we will walk through workbooks on:
  - MLPs
    - Maybe a little on autoencoders
  - CNNs
  - RNNs
- Hopefully we will have a bit of time for you modify the code and even create and run some code yourself
- Decide how you want to access the workbook. These slides will be available on the Indico agenda if you want to review the options again at a later date. Access to monty.stfc.ac.uk will be available all week