HLS4ML Environment Set Up Tutorial

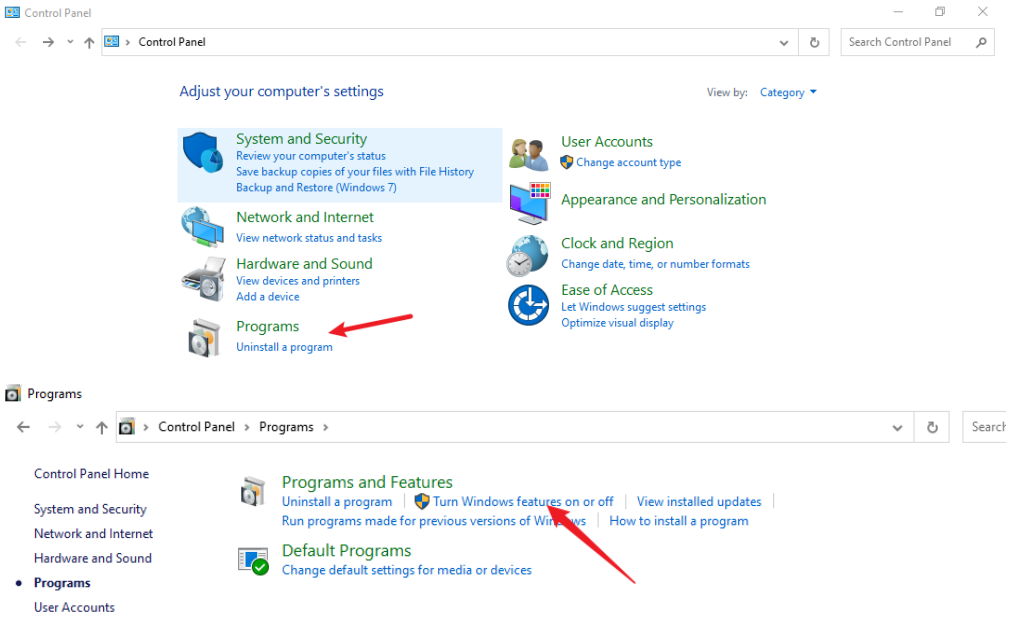
First of all, welcome to this team. This tutorial will guide you on setting up the environment you need to run HLS4ML on your own computer.

Step 1:

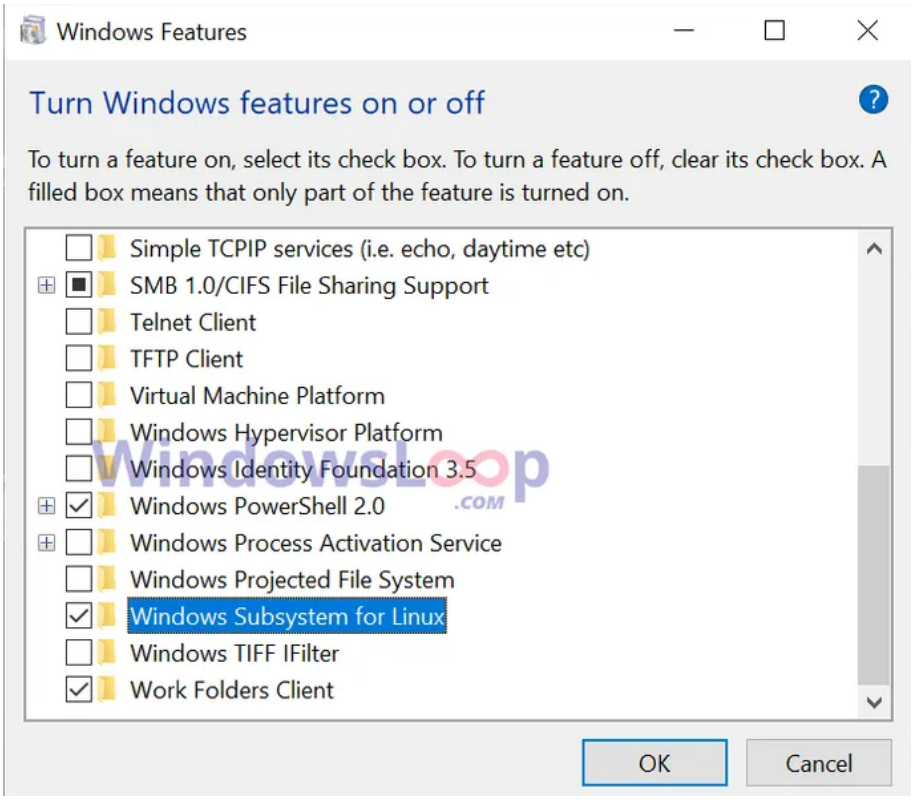
We recommend using Linux as the OS, so if you are using Linux or have Linux virtual machine on your computer, you may continue from step 2. If you are using Windows as your OS, we highly recommend using a feature called Windows Subsystem for Linux, aka WSL to run Linux. You may also build a virtual machine by yourself, but this tutorial will only teach you how to use WSL. (We currently don't know how to build environment on MacOS, if you are using MacOS, please contact Professor or Elham for help)

1. In the taskbar, click the tool and type in "Turn Windows Feature On or Off",

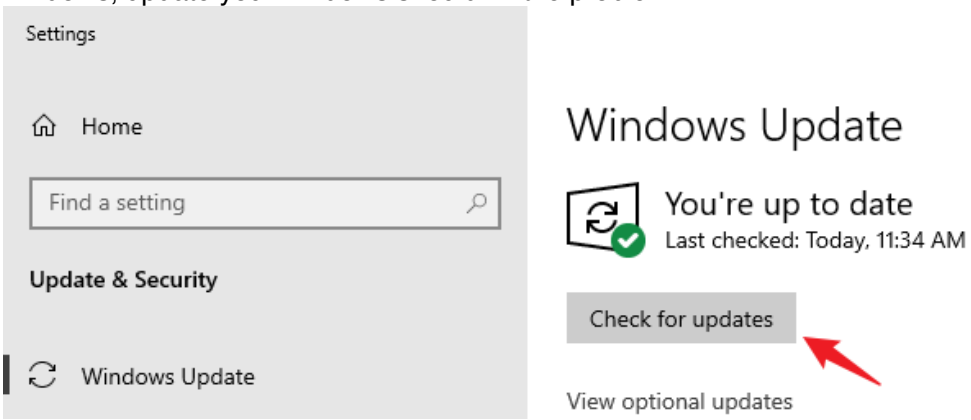
you may also find it from Control Panel -> Programs -> Turn Windows features on or off



Then select the Windows Subsystem for Linux and click OK.



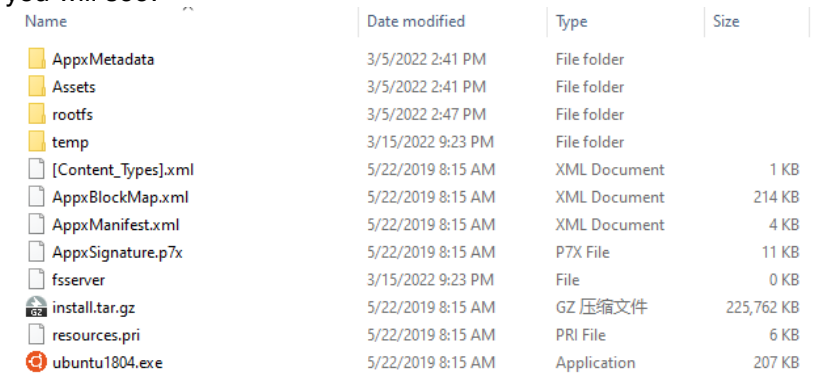
You may have some error code, generally it is due to old version of your current windows, update your windows should fix the problem.



2. Now you have enabled the feature, it is time to get a Linux for you. We recommend using Linux 18.04 LTS (Long term support version) as your Linux version. There are two ways to do so. You may download it from Microsoft Store, just search Linux 18.04 LTS.

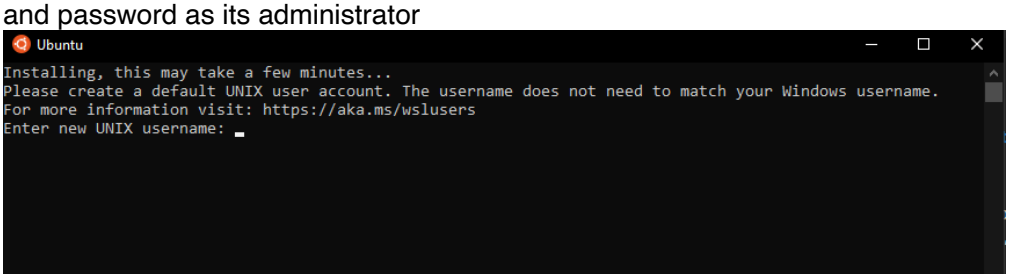
   If you don't want to install it in your C disk, you may download it from this link https://drive.google.com/file/d/1yBP0fodQXuzFd9DRWxg6J7Xh2v59tvY6/view?usp=sharing

to download to the place you want (which means you may install this on a flash drive or a portable hard drive) Once you have the zip file, unzip it to a folder and you will see:



Name	Date modified	Type	Size
AppxMetadata	3/5/2022 2:41 PM	File folder	
Assets	3/5/2022 2:41 PM	File folder	
rootfs	3/5/2022 2:47 PM	File folder	
temp	3/15/2022 9:23 PM	File folder	
[Content_Types].xml	5/22/2019 8:15 AM	XML Document	1 KB
AppxBlockMap.xml	5/22/2019 8:15 AM	XML Document	214 KB
AppxManifest.xml	5/22/2019 8:15 AM	XML Document	4 KB
AppxSignature.p7x	5/22/2019 8:15 AM	P7X File	11 KB
fsserver	3/15/2022 9:23 PM	File	0 KB
install.tar.gz	5/22/2019 8:15 AM	GZ 压缩文件	225,762 KB
resources.pri	5/22/2019 8:15 AM	PRI File	6 KB
ubuntu1804.exe	5/22/2019 8:15 AM	Application	207 KB

Then click the ubuntu1804.exe or open it from Microsoft Store or Quick Start Menu if you downloaded it from there. A terminal will pop up and Linux will start its initial set up, it will take a few minutes. Then it will ask you to set a username and password as its administrator



Please set your username with a lowercase alphabet start and no space between each word. Then type in

sudo apt update

then

sudo apt upgrade

to upgrade your Linux. (sudo means super user do, Linux will ask your password to processed the process, and when you type in your password, it will not appear on the terminal, don't worry, it is designed to do so, just type in your password and press ENTER) After this step, your Linux OS is all set.

Step 2:

As an engineer, you may be doing multiple projects at the same time, each project will use different environment and packages as its requirement. How to keep one's environment and packages just to its own? It is time to introduce "Conda", a powerful package manager and virtual environment builder.

1. In your windows browser, access https://docs.conda.io/en/latest/miniconda.html#installing
2. Under Linux installers,
   right click Python3.9 version and copy the link address
   then in your Linux terminal, type wget, then paste the link to terminal (In terminal, right click is paste), so the entire command should looks like
   "wget https://repo.anaconda.com/miniconda/Miniconda3-py39_4.11.0-Linux-x86_64.sh"
   This command will help you download Miniconda3 installation program to your Linux. Once it is finished downloading, type "bash Miniconda3-py39_4.11.0-

Linux installers

Linux

Python version	Name	Size	SHA256 hash
Python 3.9	Miniconda3 Linux 64-bit	72.2 MiB	4ee9c3aa53329cd7a63b49877c0b
	Miniconda3 Linux-aarch64 64-bit	74.4 MiB	00c7127a8a8d3f4b9c2ab3391c66
	Miniconda3 Linux-ppc64le 64-bit	73.5 MiB	8ee1f8d17ef7c8cb08a85f7d858b
	Miniconda3 Linux-s390x 64-bit	57.1 MiB	1faed9abecf4a4ddd4e0d8891fc2
Python 3.8	Miniconda3 Linux 64-bit	71.7 MiB	4bb91089ecc5cc2538dece680bfe
	Miniconda3 Linux-aarch64 64-bit	63.6 MiB	607549f9f9c5c703be850fa3025e
	Miniconda3 Linux-ppc64le 64-bit	65.2 MiB	2f606bd65ffe76a7866bc445d961
	Miniconda3 Linux-s390x 64-bit	89.0 MiB	55f514110a50e98549a68912cbb6
Python 3.7	Miniconda3 Linux 64-bit	98.9 MiB	745c99af2cb0d0e0f43c7ed1a341
	Miniconda3 Linux-aarch64 64-bit	100.9 MiB	736bd228d336f4b2d16cdc94f2e6
	Miniconda3 Linux-ppc64le 64-bit	101.0 MiB	041ba0d993398200b3e7f88aee86
	Miniconda3 Linux-s390x 64-bit	84.1 MiB	7ab9f813dd84cb0951a2d755cd84

Linux-x86_64.sh" or just "bash Mini" then press Tab, it will auto fill the rest of the command. Then follow the guidance and finish installation. Once you finished installation, terminal will ask to whether to initialize conda or not, type yes and press ENTER.

3. Now we need to edit one file to make sure your Linux recognizes the conda you just installed. You will use VIM, a powerful text editor, to edit bash_profile.

   Here are some commands you need to know for later operation:

   vim <Filename> ---to open the file we want
   :wq ---save edited file
   :q! ---quit the file without saving
   i ---change the current mode into insert mode

   Once you enter the vim, you need to press ESC before entering the command

   Please follow the steps to make the modification for your terminal:

   a) Type "vim ~/.bash_profile" and press ENTER.
   b) Press i to change the vim into insert mode.
   c) Copy and paste the following things into the terminal. **Change the <YOUR USERNAME > part into your username that you set in the step 1**.

      if [-f ~/.bashrc]; then . ~/.bashrc; fi

      export PATH="/home/<YOUR USERNAME>/miniconda3/bin:$PATH"

      alias jupyter-notebook="/home/<YOUR USERNAME>/miniconda3/envs/hls4ml-tutorial/bin/jupyter-notebook --no-browser"

   d) Press ESC and type ":wq"

e) Once you exit vim, run "source ~/.bash_profile" and "source ~/.bashrc" in your terminal

4. After making the configuration, it is time to download hls4ml into your Linux, Run the following command:

   git clone https://github.com/fastmachinelearning/hls4ml-tutorial.git

   to download the repository to your Linux, when it is finished, you need to change directory into the folder you just downloaded, here are some commands for Linux to check folders you have and change directory:

   ls ----list files in the current directory
   cd ----change directory

   a) Run "cd" to back to the home directory.
   b) Run "cd hls4ml-tutorial" to change the directory into the folder you just downloaded

   c) Run "conda env create -f environment.yml" to let conda create the virtual environment for this project.
   d) Once the environment is created, run "conda activate hls4ml-tutorial" to activate the environment. If you activate the environment correctly, you should see (hls4ml-tutorial) at the left of your terminal user command line

```
(hls4ml-tutorial) dennis_yin@DESKTOP-M7A1L8B:~$
```

5. Now you have activated the environment, you may use "conda deactivate" to exit the environment, and use "conda env list" to check how many environments you have on this Linux. Remember to activate your environment every time you want to do this project on your local machine. We now need to use Jupyter-notebook (A web-based interactive computing platform) to run all the stuffs we need. Normally, we need to have a browser to do so, but WSL doesn't have graphic interface, so we will use windows' browser to assist us. Remember the last line that we put in your bash_profile?

```
if [ -f ~/.bashrc ]; then . ~/.bashrc; fi

export PATH="/home/<YOUR USERNAME>/miniconda3/bin:$PATH"

alias jupyter-notebook="/home/<YOUR USERNAME>/miniconda3/envs/hls4ml-tutorial/bin/jupyter-notebook --no-browser"
```

   This line creates an alias that whenever you run jupyter-notebook, it will run it without showing browser, instead it will give us a token that can be access though any browser which is sharing the same local area network. Here is an example:

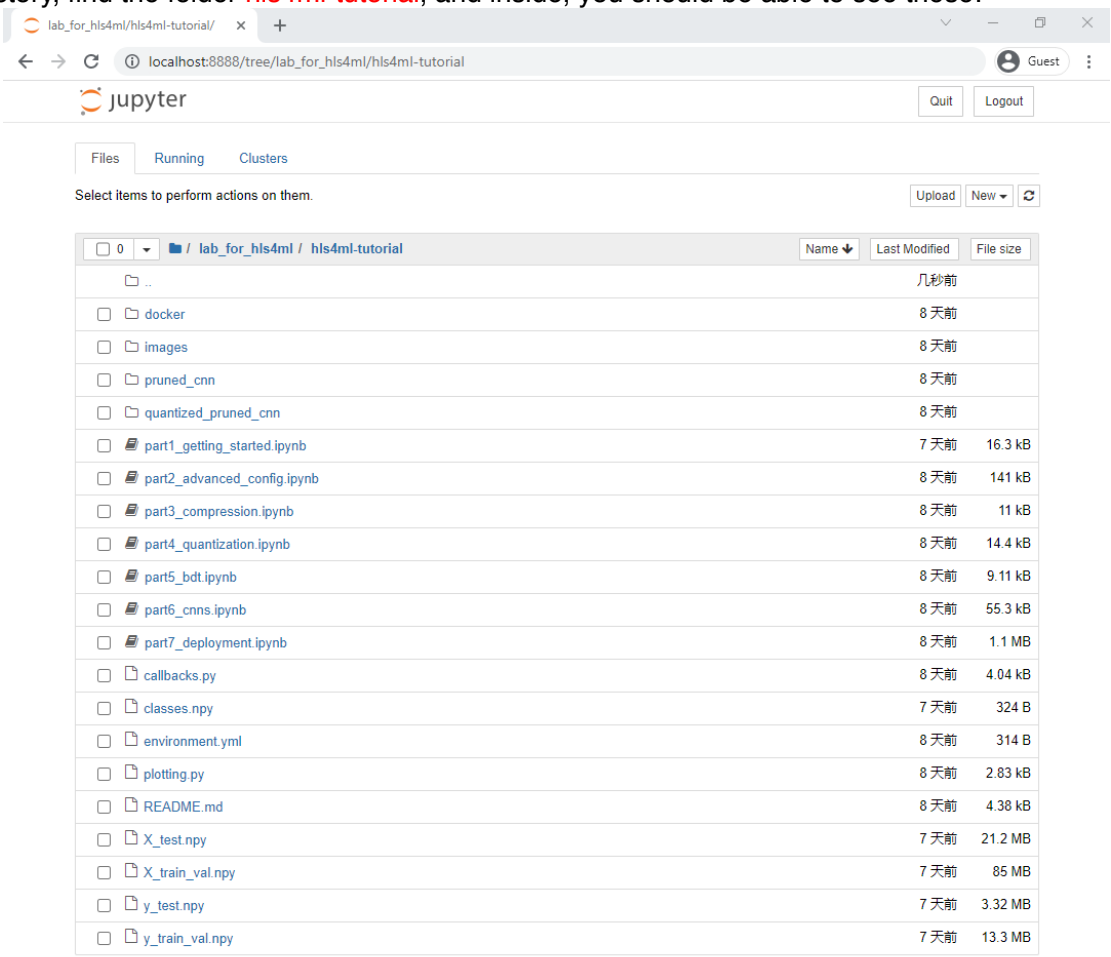
```
(hls4ml-tutorial) dennis_yin@DESKTOP-M7A1L8B:~$ jupyter-notebook
[I 14:41:24.972 NotebookApp] Serving notebooks from local directory: /home/dennis_yin
[I 14:41:24.973 NotebookApp] Jupyter Notebook 6.4.8 is running at:
[I 14:41:24.973 NotebookApp] http://localhost:8888/?token=27860f7ce6303ca9fe33d978f0bbe1fd0748541d38cef36b
[I 14:41:24.973 NotebookApp]  or http://127.0.0.1:8888/?token=27860f7ce6303ca9fe33d978f0bbe1fd0748541d38cef36b
[I 14:41:24.974 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 14:41:25.013 NotebookApp]

    To access the notebook, open this file in a browser:
        file:///home/dennis_yin/.local/share/jupyter/runtime/nbserver-1406-open.html
    Or copy and paste one of these URLs:
        http://localhost:8888/?token=27860f7ce6303ca9fe33d978f0bbe1fd0748541d38cef36b
     or http://127.0.0.1:8888/?token=27860f7ce6303ca9fe33d978f0bbe1fd0748541d38cef36b
```

   Copy one of these URL into the browser, and you may access your directory in your browser.

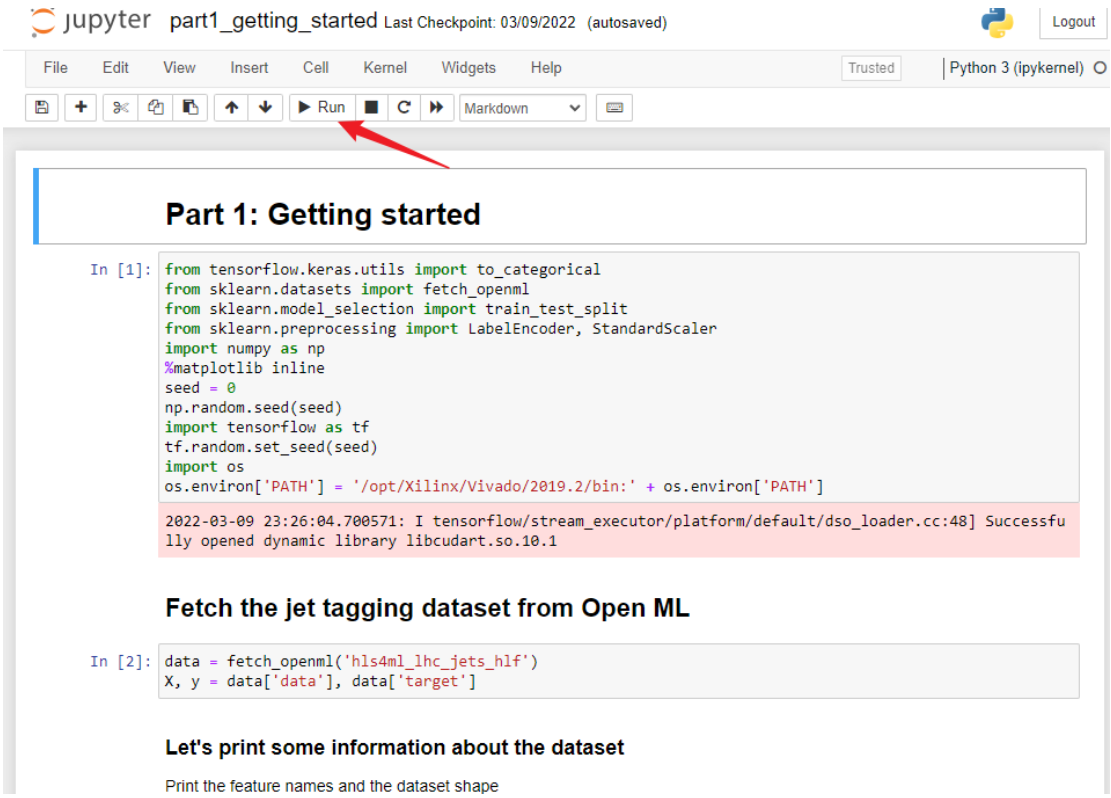
Step 3:

If everything goes smoothly, you should look something like this

This is my linux home directory, yours definitely going to looks different. In your directory, find the folder hls4ml-tutorial, and inside, you should be able to see these:



Click part1_getting_started.ipynb and run the Part 1 code, you may have some error saying not able to access dynamic library, it just means it cannot detect your GPU and it will continue process with your CPU.



Once it finished running, go back to the directory list and check whether a folder called “scikit_learn_data” has been created. If it does, congratulations! You are all set. Below are some links that could help you better understand the hls4ml and Linux terminal.

Linux Tutorial for Beginners: Introduction to Linux Operating System
https://www.youtube.com/watch?v=V1y-mbWM3B8

HLS4ML Tutorial
https://github.com/fastmachinelearning/hls4ml-tutorial

HLS4ML Guide
https://fastmachinelearning.org/hls4ml/