Docker for Windows

March 2021

1 General setup

1.1 Install Docker Desktop

Install the docker desktop according to your windows edition. The windows home edition requires an additional install of W2LS, which will be taken care by the installer itself. Make sure to have checked the option for W2LS installer in the start screen for docker desktop installer.

A restart for the machine will be prompted once docker desktop is installed (note : don't worry if W2LS is not installed yet). Once the machine restarts and you start docker desktop, a prompt for installing W2LS will be thrown if it isn't installed yet. Follow the instructions and get that installed as well. Make sure to have Docker Desktop running in the background, it can be checked by the presence of it's icon in the tray of icons on the taskbar.

1.2 Editor Note

If you are using VScode for editing files which will be executed inside the container, it is imperative that you have your default EoL set as LF instead of CRLF. This can be changed at the bottom right of the VScode editor window.

1.3 Files Description

The files in windows require a .bat extension to run unlike linux files which can run without extensions or with .sh extensions (shell scripts). These .bat files should not usually have any comments in them (# doesn't work in these files), however very necessary comments can be made using the REM command. The docker setup will be done using 3 files.

1. env.bat : is for setting up environment variables
2. Dockerfile : is for directing all required installs in the container
3. entrypoint.sh : directs the container's entrypoint

The docker-shell.bat is the only file that needs to be executed by the user, it calls all the above files in proper order and starts the container.

2 Environment File

This file is used for setting up environment variables, mainly the IMAGE_NAME and BASE_DIR variables. Windows uses SET command for assigning values to environment variables. While assigning values to IMAGE_NAME is a relatively simple task, BASE_DIR is surprisingly difficult. Taking a look at the file structure for neurodiffeq library, these docker files are present in the docker folder but when we enter the container we would want to mount the neurodiffeq folder which is one level higher than the docker folder. While in linux a simple /.. would suffice this cause, in windows you need to actually loop through the directories and get the absolute address till one level higher. More on this can be found over here.

Since this is too cumbersome a process and we don't really mind switching directories while assigning environment variables, I went one level higher, assigned the current directory and then switched back to current level. The following commands do this :

- cd ..
- assign BASE_DIR to %cd%
- cd docker

3 Dockerfile

This is the file that directs all the installs which need to be present in the container environment. Since the container is linux based we get the general apt-get packages and install python with pipenv. The only change in this file for windows is to specify the entrypoint as a .bat file instead of a .sh file

4 Entrypoint File

This is supposed to be the starting point for the docker container. This file lets the user know that container is running by displaying a message and starts up a pipenv shell.

5 Docker shell file

This is the file which gets executed and calls all the other files. It sets the environment variables by calling env.bat. Then it builds a docker image of IMAGE_NAME using Dockerfile. Next it runs the container, and it also mounts the volume BASE_DIR to the container. This is a bind mount and again an absolute path which is one level higher is required to be passed as the source. For this we do the same thing as was done in the environment file. Execute cd .. and get the current directory. Once the container is setup it starts running by executing the entrypoint file.