

Python via Conda, Vagrant, and Virtual Box

Piyanat Kittiwisit

January 14, 2021

Contents

1	Install Virtual Box	2
2	Install Vagrant	2
2.1	Install Vagrant Manager	2
3	Setup Ubuntu 20.04 (Focal Fossa)	2
3.1	Download Ubuntu Vagrant box	2
3.2	Initialise Vagrant machine	2
3.3	Connect to Vagrant machine	2
4	Install conda	2
4.1	Download Miniconda3 installer.	2
4.2	Execute the installation.	3
5	Install Python	3
6	Post installation setup (optional but recommended)	3
6.1	Set up port forwarding for Jupyter lab	3
6.2	Set up shared directory.	3
6.3	Configure Pycharm to use Python from the vagrant machine	3

conda provides an ability to create and manage multiple environments of Python executable. Each environment is completely self-contained and different module can be installed into the environment.

Vagrant is an open-source software product for building and maintaining portable virtual software development environments (virtual machine (VM)). It simplifies the configuration and management of the virtual machine, as well as enabling users to connecting to virtual machine with simple command lines.

Virtual Box is

In this setup, Vagrant is used to install, manage, and connect to an Ubuntu VM provided by Virtual Box. Then, Python is installed on the Unbuntu VM through **conda**.

Important Notes

- In this document:
 - `user@local:~$` indicates the command prompt of your local machine.
 - `vagrant@ubuntu-focal:~$` indicates command prompt of the vagrant virtual machine
- On window, the command prompt can be accessed by launching `Git Bash` app or `cmd` app.
- `Git Bash` (optional) from [here](#))
- On window, you may also need to install:
 - `ruby` from <https://rubyinstaller.org/>.
 - `perl` from <https://www.activestate.com/products/perl/>.

1 Install Virtual Box

First download and install Virtual Box from its official website <https://www.virtualbox.org/wiki/Downloads>.

2 Install Vagrant

Download and install Vagrant from its official website <https://www.vagrantup.com/>.

2.1 Install Vagrant Manager

This tool is optional but recommended. It provides GUI interface to the Vagrant commands. The installer can be downloaded from the official website of the project <https://www.vagrantmanager.com/>.

3 Setup Ubuntu 20.04 (Focal Fossa)

3.1 Download Ubuntu Vagrant box

Download the Ubuntu vagrant box with `vagrant box add`. This box file is an Ubuntu OS image that Vagrant will manage. The actual file is kept hidden from the users. The command connects to the Vagrant Cloud site (<https://app.vagrantup.com/boxes/search>) to find a matching box. Many varieties of OS are available.

```
user@local:~$ vagrant box add ubuntu/focal64
```

3.2 Initialise Vagrant machine

Initialise a Vagrant machine with `vagrant init`. The initialisation creates the **Vagrantfile**, which stores user configurable settings for the machine, and a hidden **.vagrant** directory that hosts the low-level files needed by Vagrant to communicate with Virtual Box. Thus, a new directory should be created for each initialisation of a Vagrant machine to have the configuration files contained in a single directory.

```
user@local:~$ mkdir ubuntu-focal64
user@local:~$ cd ubuntu-focal64
user@local:~$ vagrant init ubuntu/focal64
```

3.3 Connect to Vagrant machine

Start the vagrant machine with `vagrant up` and log in to the OS with `vagrant ssh`. This command can only be executed inside the directory with the **Vagrantfile** and **.vagrant** configuration files. Also note that your username inside the Vagrant machine is “vagrant” with a home directory **/home/users/vagrant**.

```
user@local:~$ vagrant up
user@local:~$ vagrant ssh
```

4 Install conda

You can now install **conda** and Python in the usual way as in a Linux OS. Here, we will install Miniconda, which provides a free minimal installer for **conda**.

4.1 Download Miniconda3 installer.

```
vagrant@ubuntu-focal:~$ wget
https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
```

4.2 Execute the installation.

Execute the bash script and follow command prompt to perform the installation. It is recommended to use `/home/vagrant/miniconda3` for the installation path. Answer “yes” when asked “Do you wish the installer to initialize Miniconda3”.

```
vagrant@ubuntu-focal:~$ bash Miniconda3-latest-Linux-x86_64.sh
```

To make the `conda` command available, you must exit out of the vagrant machine and `ssh` back.

```
vagrant@ubuntu-focal:~$ exit
user@local:~$ vagrant ssh
```

Notice that the command prompt now has `(base)` prefix, indicating that the default `base` conda environment is activated, and the `conda` command is now available.

```
(base) vagrant@ubuntu-focal:~$ which conda
/home/vagrant/miniconda3/bin/conda
```

5 Install Python

It is highly recommend that a new `conda` environment is built every project, test, development, and etc.

Here, an environment named `astro` is created with Python 3.8 and some commonly used modules in data science and astronomy.

```
(base) vagrant@ubuntu-focal:~$ conda create -n astro -c conda-forge python=3.8 numpy scipy
matplotlib astropy pandas xarray dask h5py netcdf4 h5netcdf healpy jupyterlab ipykernel
ipymp1 nodejs
```

The `astro` environment must be activated for it to take effect.

```
(base) vagrant@ubuntu-focal:~$ conda activate astro
(astro) vagrant@ubuntu-focal:~$ which python
/home/vagrant/miniconda3/envs/astro/bin/python
```

See `conda` documentation (<https://docs.conda.io/en/latest/>) for further usages of the `conda` command.

6 Post installation setup (optional but recommended)

6.1 Set up port forwarding for Jupyter lab

If using Jupyter lab (or Jupyter notebook but you should really be using Jupyter lab), a port forwarding can be set up to allow using the local web browser to connect to a Jupyter server running inside the vagrant machine.

- `jupyter lab --generate-config`
- Note that you will have to bind Jupyter to 0.0.0.0 ip (`jupyter lab --ip=0.0.0.0`) to have Vagrant forward the port correctly. (See [here](#) and [here](#)).

6.2 Set up shared directory.

6.3 Configure Pycharm to use Python from the vagrant machine

If using Pycharm, follow [the instructions in this link](#) to use Python from the vagrant machine as an interpreter.