1. Install ipywidgets

1.1 Ensure you have successfully installed JupyterLab and can use it normally.

1.2 View whether node and nmp is installed. If the version number is displayed, it means it has been installed.

You can continue to operate (the part 1.3 of installing Node.js can be skipped directly), otherwise you need to install Node.js.

node -v && npm -v

```
pi@raspberrypi:~ $ node -v && npm -v
v12.19.0
6.14.8
```

```
pi@raspberrypi:~ $ node -v && npm -v
v10.21.0
5.8.0
```

1.3 Install **@jupyter-widgets/jupyterlab-manager** (this step needs to be operated in jupyter lab)

First, we need to enable third-party extended community.

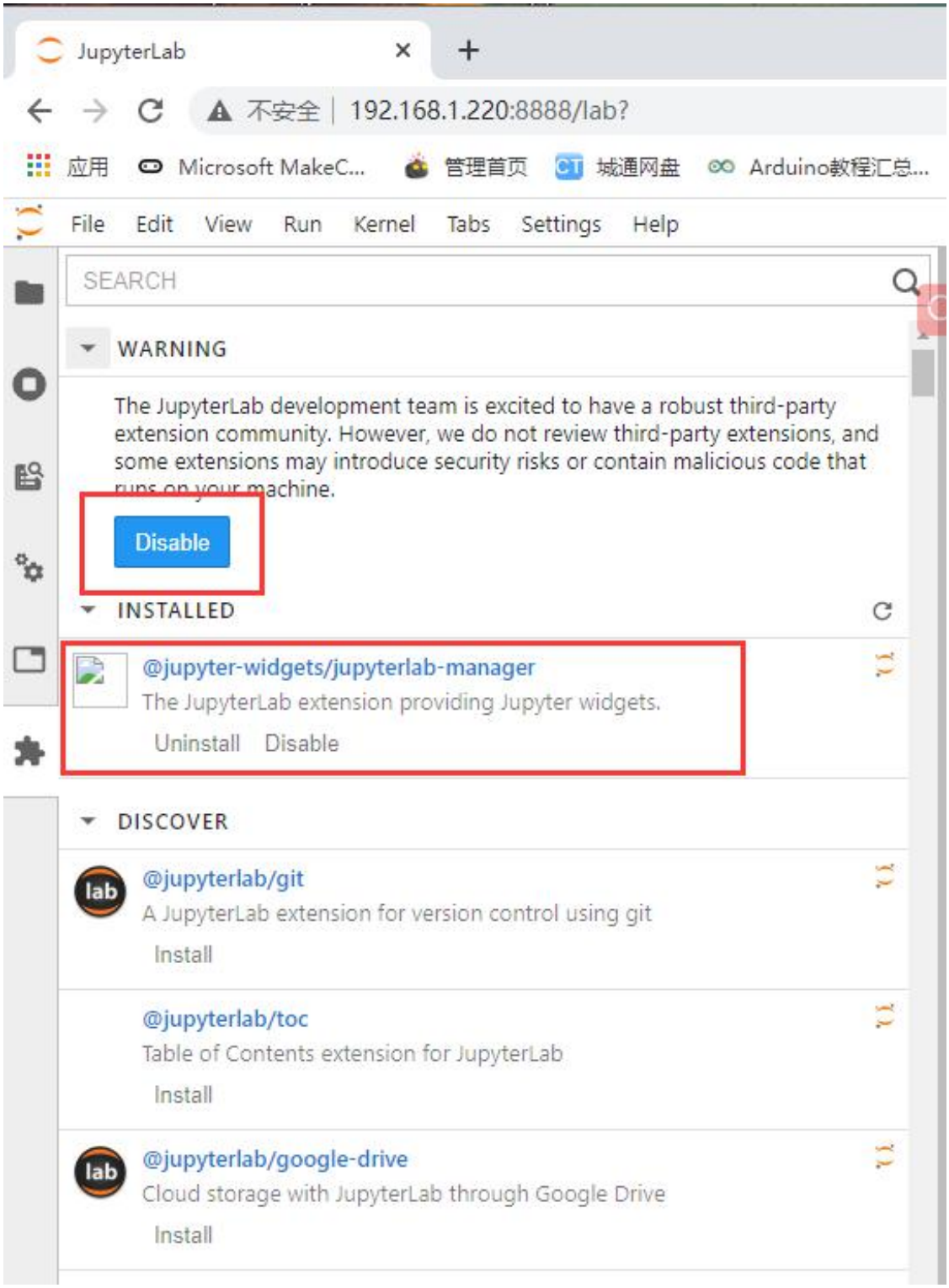
Then, install (install)**@jupyter-widgets/jupyterlab-manager** management plugin

After installation, the plug-in will appear in the INSTALLED (installed) area, as shown below.

In generally, you will be prompted to rebuild JupyterLab after the installation is complete.

This process will be relatively long, and there will be no prompt after it is installed successful.

Wait about 2-3 minutes before re-entering JupyterLab. If there is no prompt to Build JupyterLab, it means that the Build was successful.



1.4 Install ipywidgets

sudo pip3 install ipywidgets

1.5 Start up widgetsnbextension

jupyter nbextension enable --py widgetsnbextension

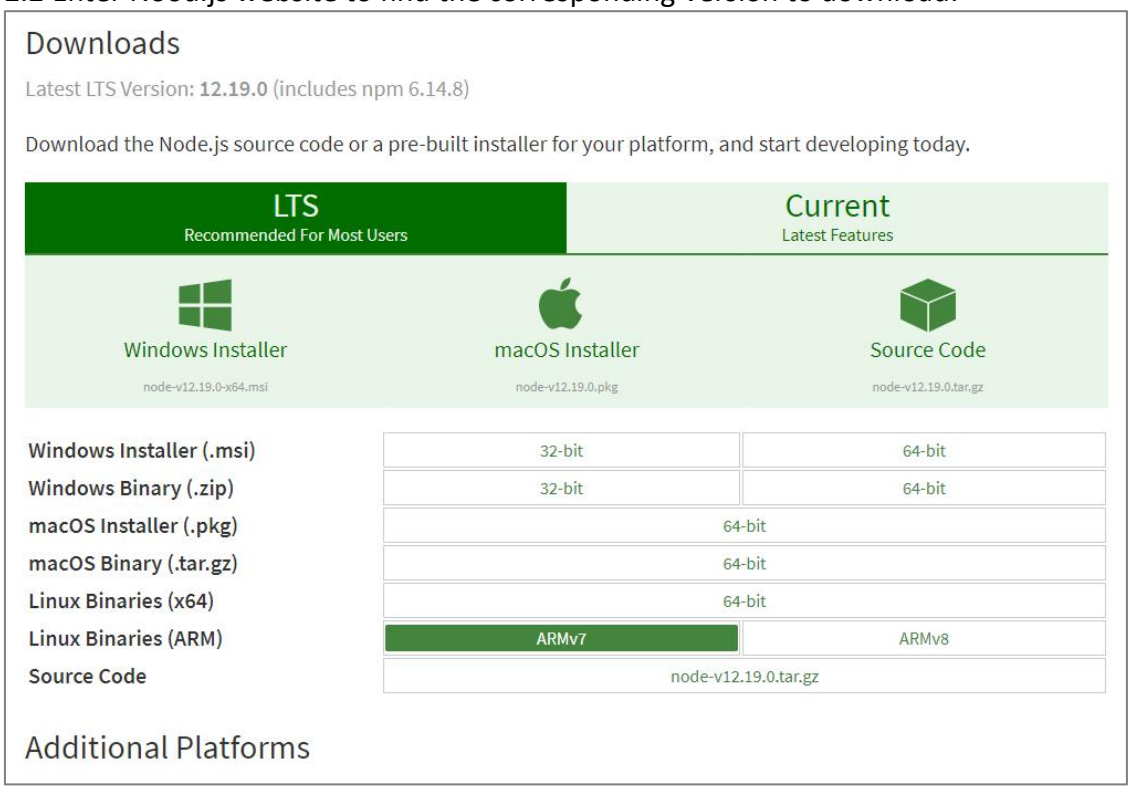
1.6 Delete temporary and static directories
jupyter lab clean
jupyter lab path

1.7 Restart Raspberry Pi
sudo reboot

2. Install Node.js
2.1 Input following command to check the architecture of the Raspberry Pi, as shown below, you can find that my Raspberry Pi 4B is armv7.
uname -a

```
pi@raspberrypi:~ $ uname -a
Linux raspberrypi 5.4.51-v7l+ #1333 SMP Mon Aug 10 16:51:40 BST 2020 armv7l GNU/Linux
```

2.2 Enter Nood.js website to find the corresponding version to download.



2.3 Extract the downloaded compressed file (the current Nood.js official website provides version 12.19.0, if the subsequent version changes, please refer to the actual situation)
xz -d node-v12.19.0-linux-armv7l.tar.xz
tar -xavf node-v12.19.0-linux-armv7l.tar

2.4 Delete the original /usr/bin.node in the system.
sudo rm -rf /usr/bin/node

2.5 Input following command.
sudo mv ./node-v12.19.0-linux-armv7l /usr/local/node

2.6 Input the following command to establish a soft connection between node and npm
sudo ln -s /usr/local/node/bin/node /usr/bin/node
sudo ln -s /usr/local/node/bin/npm /usr/bin/npm

3.Test
3.1 Enter jupyter lab
jupyter lab

3.2 Enter Notebook

3.3 Copy following code

```
from __future__ import print_function
from ipywidgets import interact, interactive, fixed, interact_manual
import ipywidgets as widgets
def f(x):
   return x
interact(f, x=10);
```

3.4 Press “Enter+Shift” key.
If the result shown in the figure below appears, it means that the Jupyter Widgets configuration is complete.

[1]:
```
from __future__ import print_function
from ipywidgets import interact, interactive, fixed, interact_manual
import ipywidgets as widgets
def f(x):
    return x
interact(f, x=10);
```

x 10