Install Python

## MATLAB Compatibility

See. For a list of compatible versions of python for matlab: [Versions of Python Compatible with MATLAB Products by Release - MATLAB & Simulink (mathworks.com)](https://www.mathworks.com/support/requirements/python-compatibility.html?s_tid=srchtitle_site_search_1_python%20compatibility)

## Windows (RECOMENDED)

<https://www.python.org/downloads/windows/>

(follow exe prompts using default installation suggestions, no need to extend path limit).

## Linux

Get a Linux subsystem for windows. This can be done using the **Windows App Store**. I use Ubuntu, but most Linux based OS’s will work.

Use this command to get to your windows drive: “cd /mnt/c/”

From there navigate to your python .tgz file and use this tutorial to install:

<https://medium.com/@lupiel/installing-python-from-a-tgz-file-a-step-by-step-guide-4cf5f4a17a86>

# Installing FOOOF

## Windows

I’m using python version 3.7 and MATLAB version 2020b for this example. If you use any other version, check the compatibility of python for your version of MATLAB (<https://www.mathworks.com/support/requirements/python-compatibility.html>). Make sure to keep your python version in mind while going through this tutorial. I will be using MATLAB 2021b and python3.7 .

Downloads

GitHub for the MATLAB wrapper for FOOOF: <https://github.com/fooof-tools/fooof_mat>

Helpful Links

If you are running into bugs: <https://irenevigueguix.wordpress.com/2020/03/25/loading-python-into-matlab/>

Tips

If you are having trouble getting your command prompt to point to the python version you want to work with, you may also configure your PATH variable to change how python3 command operates. HOW TO: in the windows search bar type: “edit the system Environment variables”. Click on the first search option of the same name. In the bottom right there will be a button “Environment Variables…”. Under the “System Variables” section click the “New…” button. For the variable name put “PYTHONPATH” and for the variable value put the path to your desired python version. Alternavtively you can just cd to the path of your python install and call the local python.exe and pip.exe.

Installation guide

In windows command prompt:

1. Check current version

Type “ python”, and in the python interpreter type:

>>> import os

>>> import sys

>>> os.path.dirname(sys.executable)

You should get a path output like: “ 'C:\\Users\\jsalminen\\AppData\\Local\\Microsoft\\WindowsApps\\PythonSoftwareFoundation.Python.3.11\_qbz5n2kfra8p0' ”

1. Make sure to cd to your python installation, then pip install these packages (These may or may not be required for your desired python installation, but trying to install them shouldn’t cause issues, key word being “shouldn’t”)

“python –m pip install cpython”

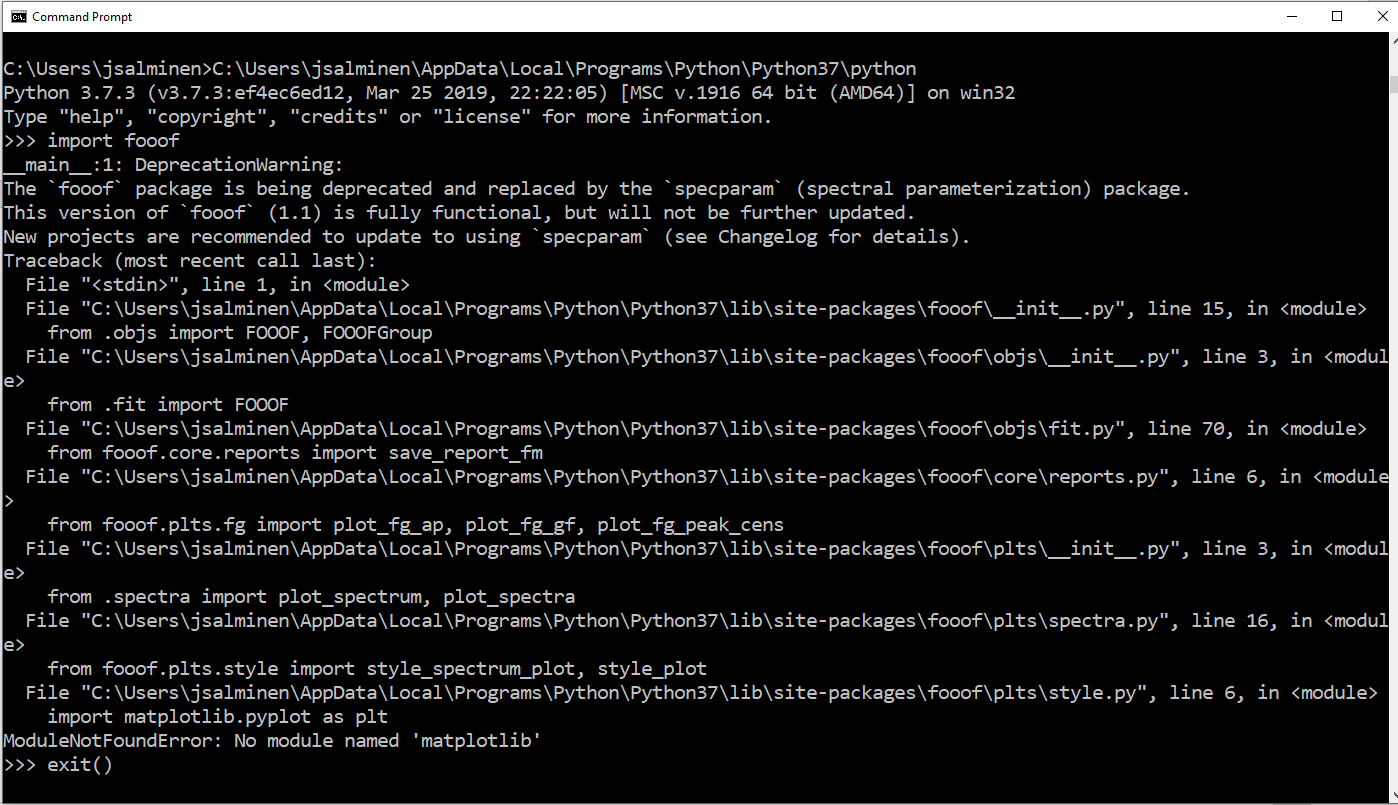
“python –m pip install numpy”

“python -m pip install matplotlib”

“python –m pip install fooof”

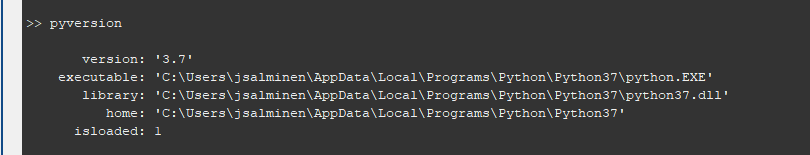
1. Double check your installation is valid. To do this, you may try opening your python interpreter in command prompt and type: “import fooof”.

Output:



If there is an error try to resolve it then move on to using the install in MATLAB. In the example above I’m missing the matplotlib library even though the fooof setup.py says I don’t need it… I will pip install matplotlib and the package should import as desired.

# Linking python to MATLAB

Type: “pyversion” 

If anything other than your desired python is loaded you will need to find where the exectuable is located and added it using the MATLAB command window and type:

“pyversion(‘path/to/my/python.exe’);”

You may need to restart matlab before and after doing this.

# Virtual Environments for the Blue Drive

(UNDER DEVELOPMENT)

Helpful Links

General HiperGator Tutorial: <https://wiki.weecology.org/docs/computers-and-programming/hipergator-intro-guide/>

Guide for getting started with Python on the HPC: <https://help.rc.ufl.edu/doc/Python>

Tips

Guide

In a HPC terminal:

1. module load python
2. cd /to/folder/where/to/install/virtualenv/
3. python -m venv ./venv
4. try to activate venv: “source ./venv/bin/activate”
5. a (venv) should appear in front of your username and current dir tag:

A black background with white text

Description automatically generated

1. check install:

>>> import os

>>> import sys

>>> os.path.dirname(sys.executable)

ERROR. If any path other than the python in your venv appears contact IT via slack or help tickets.

1. Exit the python interpreter: exit()
2. Pip install your desired packages: python -m pip install <desired package>
3. Have fun.

Recommended Packages: PyTorch, Scipy, scikit-learn, ipykernel

Setting up Visual Studio Code for Remote Development: <https://help.rc.ufl.edu/doc/VS_Code_Remote_Development>

1. Download the CLI x64 Linux version from: <https://code.visualstudio.com/#alt-downloads>
2. Copy tar.gz to folder on blue drive
3. Install using a hipergator terminal: tar -xvf vscode\_cli\_alpine\_x64\_cli.tar.gz
4. Cp code ~/bin/ (If it gives you an error that the directory doesn’t exist, make the dir using “mkdir” command
5. Start a slurm dev session: srundev
6. Start vscode tunnel: .code tunnel **or** ~/bin/code tunnel
7. You will prompted to login with a Microsoft or GitHub account. Choose the one that your VSCode install is linked to.
8. In VSCode in the bottom left click on the “Accounts” icon (looks like a little dude) then press “Enable remote tunnel access”.
9. You will then be prompted that you are connecting to the HiperGator remote session and given the option to copy a URL. Copy the URL and paste in a browser.

(THIS DOESN’T ALLOW EDITING OF CODE) Setting up your Linux Subsystem for Windows: <https://help.rc.ufl.edu/doc/Samba_Access>

1. Open your Linux subsystem via the windows app. You can also do this in visual studio code by going to “Terminal” in the top bar and clicking “New Terminal”. Now in the bottom right click the down arrow next to the “+” sign and click on your WSL install. E.g., if you are using ubuntu it will look like: “Ubuntu (WSL)”.
2. Install cifs-utils: sudo apt-get install cifs-utils
3. Make the dir to link the blue-drive to:
   1. Cd /mnt/
   2. Mkdir ./blue
   3. Mkdir ./blue/dferris
4. Link blue drive: sudo mount -t cifs -v -o user=<gatorlink username>,domain=ufad,sec=ntlmssp //exasmb.rc.ufl.edu/blue/dferris /mnt/blue/dferris
5. Activate visual studio code editing:
   1. See. <https://code.visualstudio.com/docs/remote/wsl-tutorial>
   2. Make sure you have the WSL and Remote Development extensions installed on your local windows OS.
   3. In the VS code Linux terminal: cd /to/your/python/venv/
   4. Type: code .
   5. This should install vscode in your local folder and open a vscode editor.
   6. In the new vscode editor activate your venv and have fun.

Issues

1. (FIXED) Initial issues

You would need to contact IT and make them install CPython, Matplotlib, and FOOOF onto your desired python install.

See. For a list of compatible versions of python for matlab: [Versions of Python Compatible with MATLAB Products by Release - MATLAB & Simulink (mathworks.com)](https://www.mathworks.com/support/requirements/python-compatibility.html?s_tid=srchtitle_site_search_1_python%20compatibility)

I’ve tried:

* An installation of python using the windows installer (or linux) to the location on the blue drive.
* Creating a virtual environment using a loaded python version on a hipergator terminal (e.g., module load python/3.10 >> python -m venv ./venv). Hipergator doesn’t give you access to pip libraries for installation due to security issues.

The least buggy way, and unfortunately the slowest, is to use a downloaded python version for windows/mac and run it locally.

# UPDATES