

Getting Started with Anaconda

How to download and install Anaconda on your computer, set up the Jupyter Notebook application, and install additional Python packages

Anaconda is an open source programming platform for writing Python code. Anaconda is free for individual users, and it is easy to download and install.

Contents

Section 1: Download and Install Anaconda	1
Section 2: Open Jupyter Notebook	2
Section 3: Download Additional Python Packages (Libraries)	3
Section 4: Make a New Folder in Jupyter Notebook and Upload Code Files	5

Section 1: Download and Install Anaconda

We recommend installing Python version 3.8. *If you already have Anaconda installed on your computer and it is Python version 3.7 or earlier, we suggest upgrading to Python version 3.8.*

Open [Anaconda’s webpage for individual products](#). Scroll to the bottom of the page to the “Anaconda Installers” section (Figure 1). Select the preferred installer for your computer’s operating system and click on the corresponding link to save the “Anaconda3 Installer” on your computer.

After the installer has downloaded, click on the .exe and follow the instructions to install Anaconda (Windows or MacOS Graphical Installer) or use the Command Line Installer, depending on which option you selected. During the installation setup, if you encounter the option to “Add Anaconda to my PATH environment variable” (Figure 2), do NOT select this option! It can cause problems later on when using Python IDEs and packages.



Figure 1. Screenshot of Anaconda Installers (for individuals) section on Anaconda.com

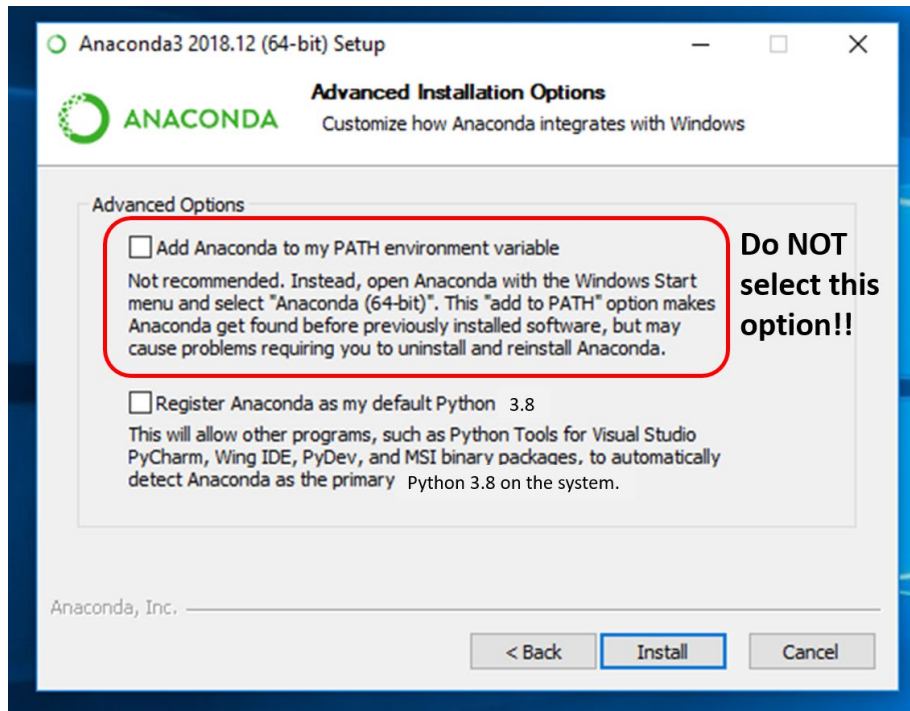


Figure 2. Options window that users may encounter during Anaconda installation

Section 2: Open Jupyter Notebook

Jupyter Notebook is a user-friendly, web-based interactive environment that makes learning Python easy.

Depending on your computer's operating system, there are a couple of ways to open Jupyter Notebook.

Open "Anaconda Navigator" and click the "Launch" box under the Jupyter Notebook application (Figure 3).

Alternatively, if you are using a Windows operating system, under "Anaconda" in the "Start" menu, there should be a link for "Jupyter Notebook."

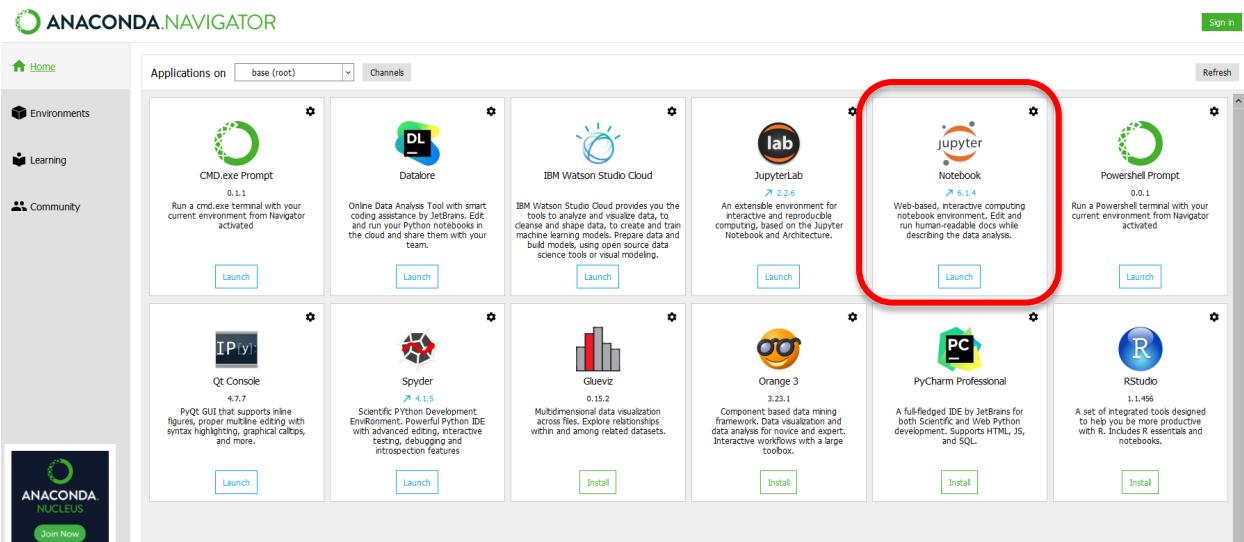


Figure 3. Screenshot of Anaconda Navigator, with Jupyter Notebook application circled in red

Section 3: Download Additional Python Packages (Libraries)

The Jupyter Notebook home screen will open in a browser window or tab (e.g., Chrome) and look similar to Figure 4.

Anaconda comes loaded with many helpful Python packages (also called libraries). To work with satellite datasets, we will use several of these preinstalled libraries (e.g., NumPy, Matplotlib). However, we need to install additional packages using the **Anaconda Powershell Prompt**.

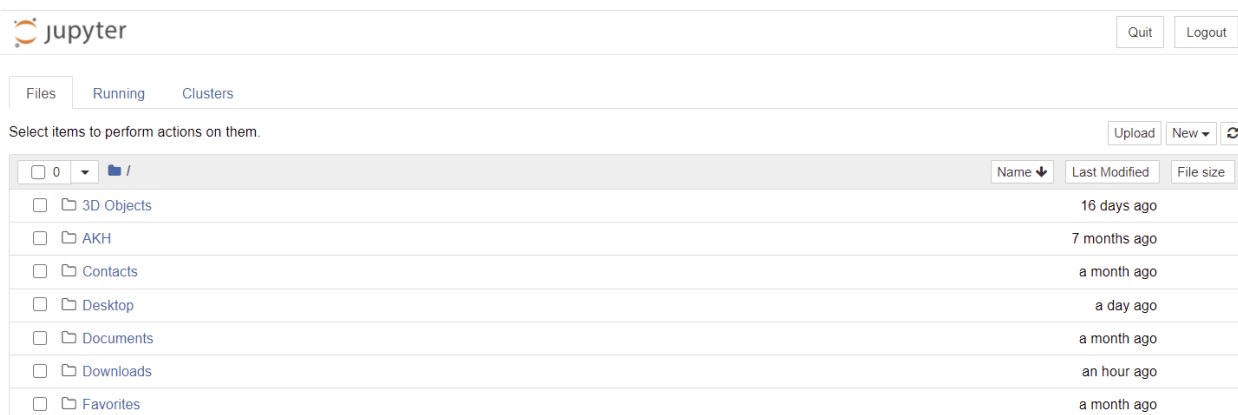


Figure 4. Example of Jupyter Notebook home screen

The easiest way to open the Anaconda Powershell Prompt is to click on the “New” drop down button, located in the upper right-hand corner of the Jupyter Notebook home screen, and choose “Terminal” (Figure 5).

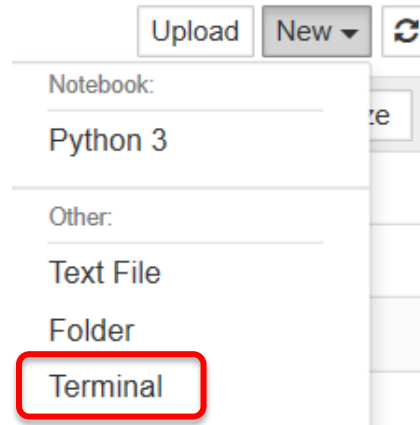


Figure 5. Screenshot of Jupyter Notebook home screen, showing how to open a Terminal

If you are using a Windows operating system, you can select “Anaconda Powershell Prompt” under “Anaconda” in the Start menu, or you can type “Anaconda Powershell Prompt” into your Windows search bar.

Install the following packages:

- **netCDF4**
- **s3fs**
- **cartopy**

In the Anaconda Powershell Prompt, type the following command into the terminal for each package and then press Enter:

`conda install package-name`

For example:

```
Anaconda Powershell Prompt (Anaconda3.8)
(base) PS C:\Users\Amy.Huff> conda install netCDF4
```

The terminal should begin preparing to install the package; you will be asked whether you want to proceed as shown:

```
Proceed ([y]/n)? y
```

Type the letter “y” and press Enter, and the package should begin downloading. Repeat this process for each of the packages listed above.

Section 4: Make a New Folder in Jupyter Notebook and Upload Code Files

The last step is to upload the Python code files we will use in the training session. Return to the Jupyter Notebook home page. Click on the “New” drop down button, located in the upper right-hand corner, and select “Folder.”

A new folder called “Untitled Folder” (Figure 6) will appear. To rename the folder, click on the little box next to the folder icon and then click the “Rename” button near the top of the page (Figure 6).

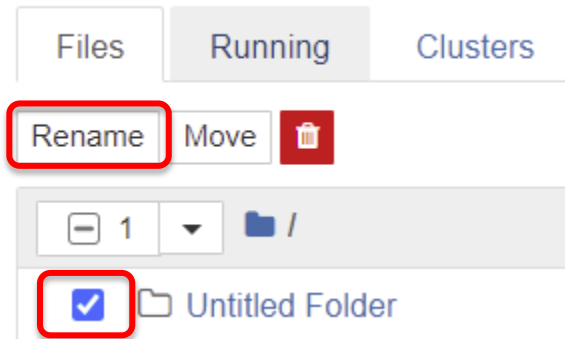


Figure 6. Screenshot of Jupyter Notebook home screen, showing how to rename a new Untitled Folder

Once the new folder has been created and renamed, open it. If you haven’t already, download the .zip file from the training website; unzip the folder and extract the contents (4 Python code files in .ipynb format). Click on the “Upload” button (located next to the “New” drop down button) and select the 4 code files. The file names will appear in the Jupyter Notebook home screen (Figure 7); click on the “Upload” button for each file to finish uploading.



Figure 7. Example of Jupyter Notebook training folder, showing how to upload files

The folder containing the uploaded files will look similar to Figure 8. To open a code file, click on the file name (e.g., “Exercise-5_ABI_Download_AWS.ipynb”).

Select items to perform actions on them.

Upload New ↕

<input type="checkbox"/>	0	AKH / Trainings / AMS-Short-Course_Mar2021	Name ↓	Last Modified	File size
		..		seconds ago	
<input type="checkbox"/>		Backup		a day ago	
<input type="checkbox"/>		BONUS_Animate-Figures.ipynb		a day ago	4.75 kB
<input type="checkbox"/>		Exercise-5_ABI_Download_AWS.ipynb		a day ago	6.57 kB
<input type="checkbox"/>		Exercise-6_ABI-AOD_Explore.ipynb		a day ago	6.67 kB
<input type="checkbox"/>		Exercise-7_ABI-AOD_Visualize.ipynb		a day ago	17.7 kB

Figure 8. Example of Jupyter Notebook training folder, containing uploaded code files