Getting Started with Anaconda: Installation and Basics

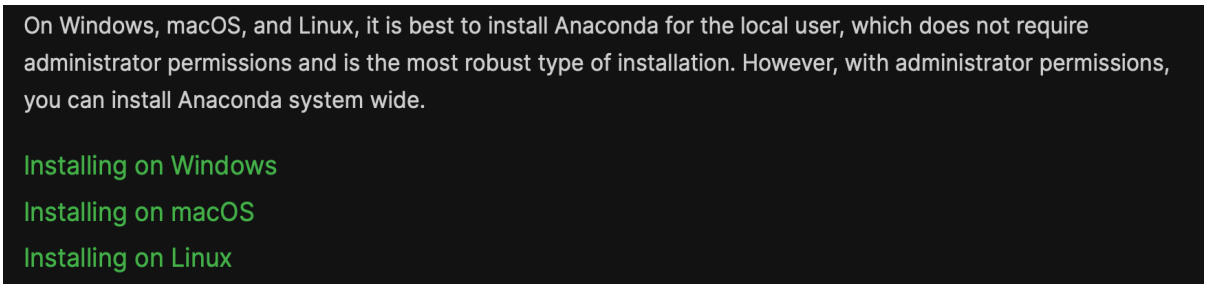
Anaconda is a powerful platform for data science and programming in Python. Anaconda installation includes python, extra python packages and Jupyter notebook.

This document will guide you through the installation and basic usage of Anaconda on various operating systems.

Section 1: Installation

1.1 Installing Anaconda:

- Download Anaconda from the official website:
  https://docs.anaconda.com/free/anaconda/install/
  Click the installation link on the website for macOS, Windows, Linux accordingly:



- Follow the installation guide for your platform on the official website. For macOS, we recommend installing using wizard, rather them command line.
- Once installed, verify the installation of the Anaconda Navigator:
  - On **Windows / Mac**, search for Anaconda Navigator in your system's applications.
  - On **Linux**, run terminal (*Ctrl* + *Shift* + *T*) and type:

    ```
    anaconda-navigator
    ```

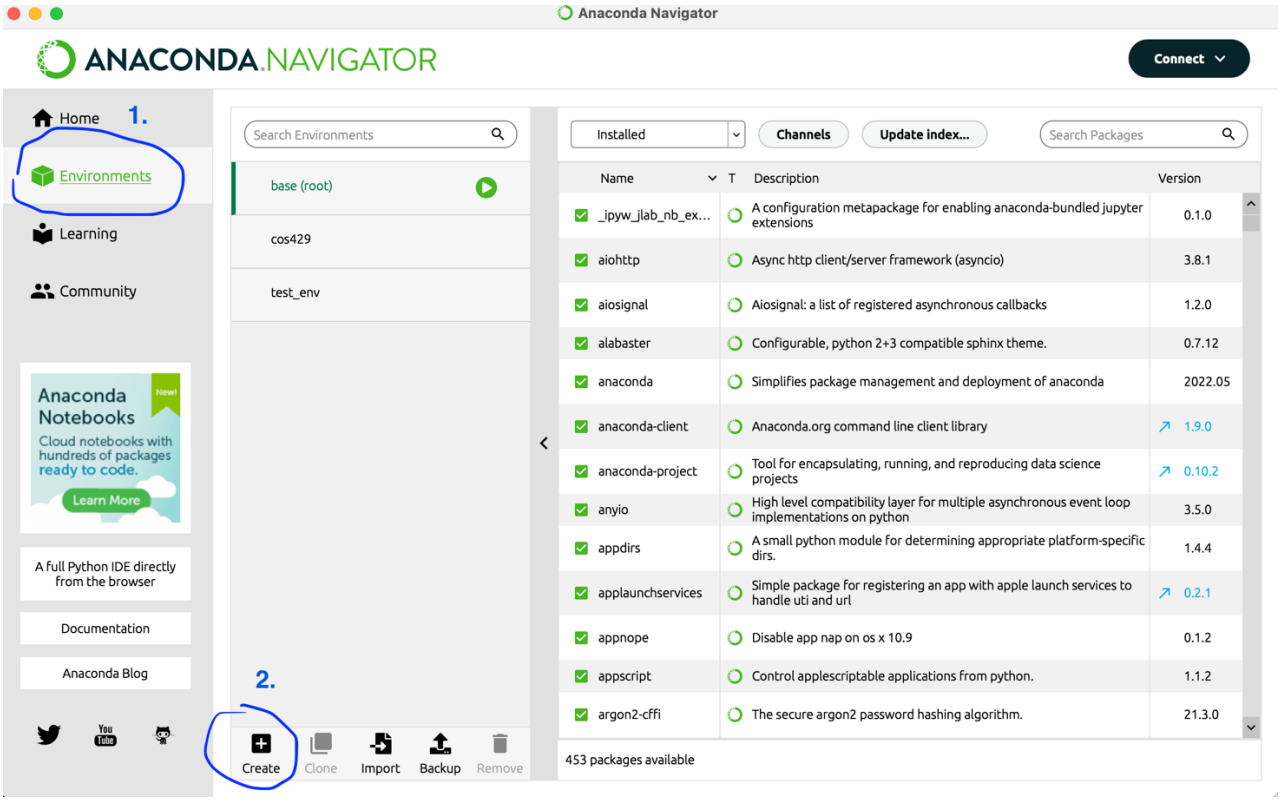
Section 2: Using Anaconda Navigator with graphical interface.

2.1 Creating a New Environment:

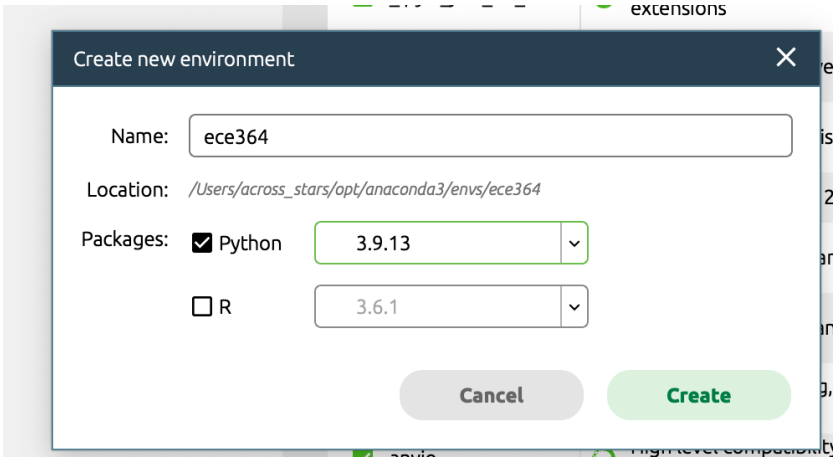
After launching Anaconda, by default, you are in ***base*** environment. We recommend creating a new environment for every major project to isolate different version of packages from each

other. E.g., for this class, you might want to create an environment ***ece364*** and install all necessary packages there:

- Open Anaconda Navigator.
- Click on the "Environments" tab.
- Click the "Create" button to create a new environment.



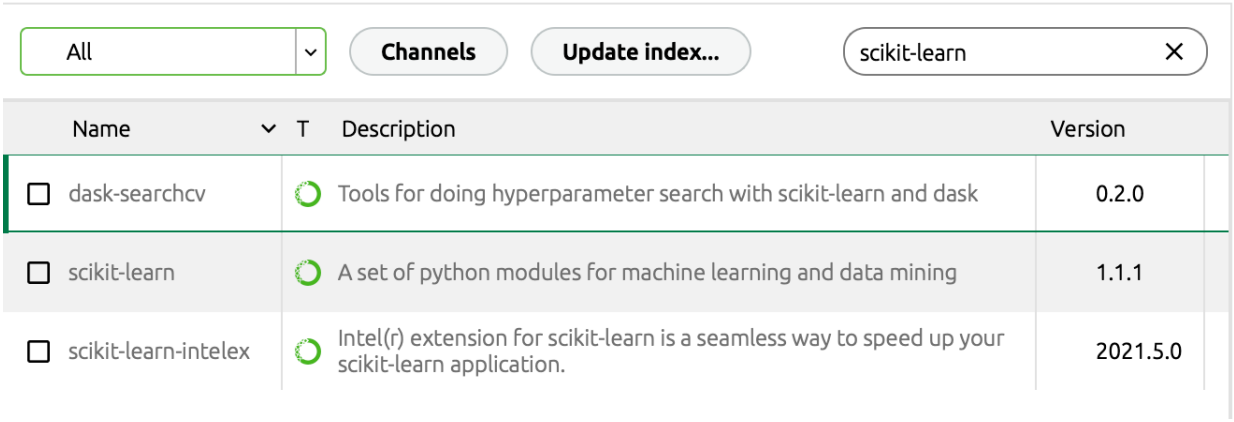
- Specify the environment name and choose the Python version.



- Click "Create" to complete the process.

2.2 Installing Packages:

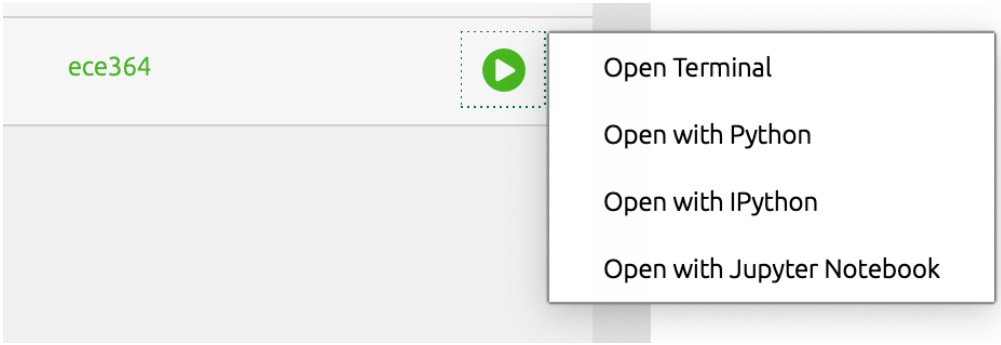
- In Anaconda Navigator, select your environment.
- Choose “All” in the drop-down menu:



- Type the name of required package in "Packages" tab to search, next select the package you need and click “Apply”. For the first assignment, you need to add the following packages into your working environment:

  Pandas, numpy, matplotlib, matplotlib-inline, seaborn, scikit-learn

- To run Jupyter, tap the play button near the activated environment and select Jupyter:



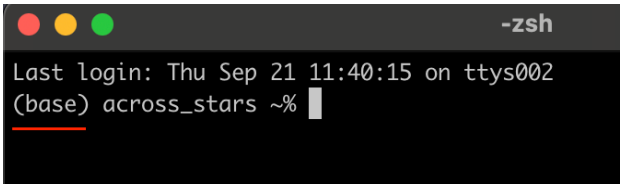
2.3 Managing Environments:

- You can activate and deactivate environments from the "Home" tab.
- Use the "Environments" tab to manage existing environments (remove them)

Section 3: Using Anaconda Navigator in the Command Line

3.1 Activating and Deactivating Environments:

- Open a terminal. By default, you are in base environment:



- Create a new environment called `ece364`:

  ```
  conda create --name ece364
  ```

- Activate the environment using the command:

  ```
  conda activate ece364_name
  ```

3.2 Installing Packages via CLI:

- Pakcages will be installed into your *active* environment, so don't forget to activate it first. For instllation commands, you may simply google the required package: *anaconda install <your package name>* and copy the commands from the official anaconda website. For the first assignment the commands will be:

  ```
  conda install -c anaconda scikit-learn pandas numpy seaborn
  ```

  ```
  conda install -c conda-forge matplotlib
  ```

3.3 Running Jupyter:

- Run jupyter in your active environment running the command:

  ```
  jupyter-notebook
  ```

  It should start in your browser. Now you can navigate the folder tree.