Virtual Environments

One of the most important tools for a Python Developer are **virtual environments**. They are a tool that help you keep dependencies required by different projects separate. I would recommend you keep different virtual environments for different projects/labs/class as some may require newer/older versions of packages/modules that may not be compatible with each other. Most likely, you've been using your global installation of Python and installing packages on it without any problems, but when you are working on some pipelines, it may be advantageous to work with virtual environments that have only the necessary packages installed on it because it is easier to document and can make your workflow reproducible/repeatable.

There are several ways to create virtual environments. I have multiple versions of Python on my Windows 10 PC and personally use the Python package **venv** to create virtual environments. Then I install Jupyter Labs and any necessary packages for the project I'm working on. But I think most of you have **Anaconda** installed for your courses, and it offers an easy way to create virtual environments. Additionally, Anaconda also comes with Jupyter Labs and Jupyter Notebook installed, allowing you to access these tutorials easily.

In this tutorial, we will create a virtual environment on Anaconda for you to use for analysis in this lab, since I think most of you have it installed. However, if you do not have it installed and are interested in creating virtual environments another way, I can help you with that personally. I work with a Windows operating system, so my help may be limited on you Mac users, but I will try my best to help. Let's get started.

Let's Check if Anaconda is Installed

1. Let's verify if you Anaconda installed and open the Anaconda Prompt
    a. For Windows, click Start and search for "Anaconda Prompt" from the menu and select it
        i. Type "conda list" and press Enter. This will list all packages and their versions
    b. For macOS, press Command and Space simultaneously to open Spotlight and type Terminal. Press Return/Enter
        i. Type "conda list" and press Return/Enter button

Creating a Virtual Environment/Installing Packages

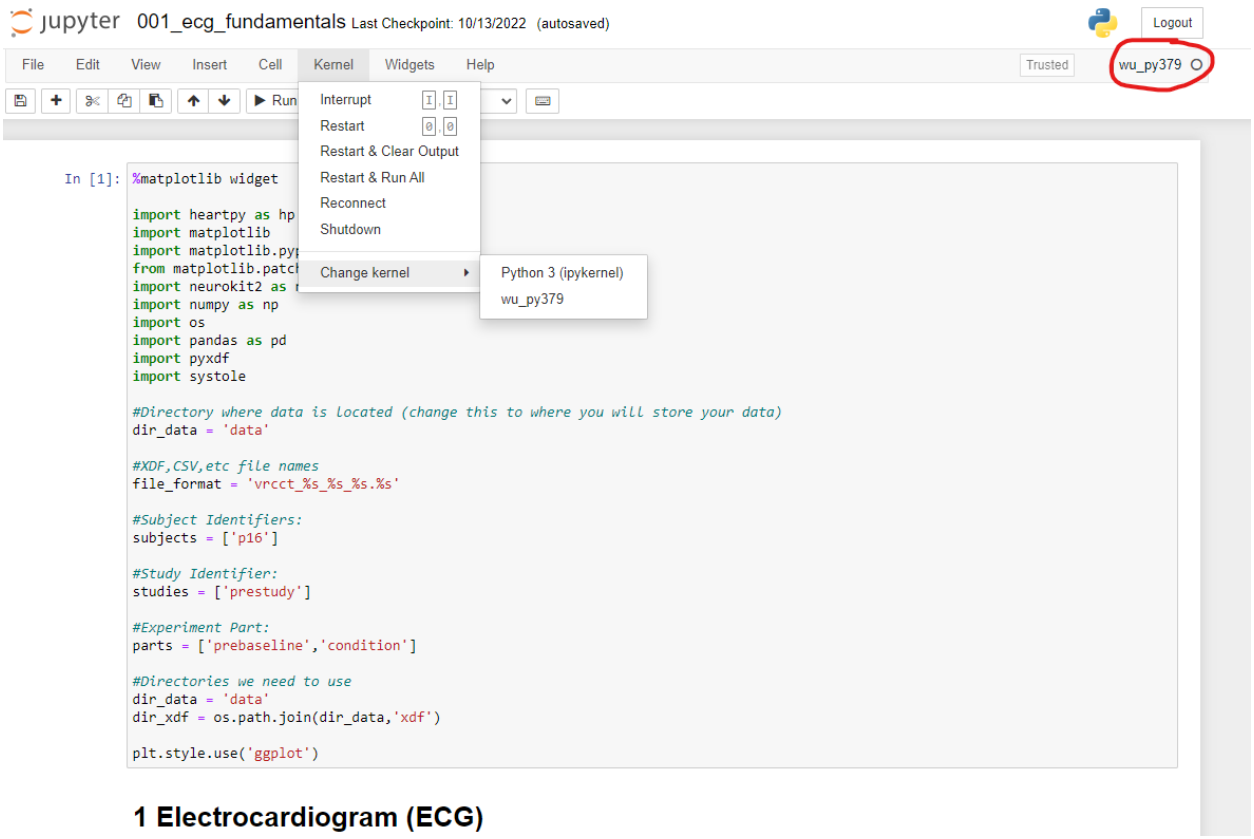
1. Open Anaconda Prompt
    a. For Windows, click Start and search for "Anaconda Prompt" from the menu and select it
    b. For macOS, press Command and Space simultaneously to open Spotlight and type Terminal. Press Return/Enter
2. Create a virtual environment using conda using our YAML file
    a. Download the YAML file from the Github Repo
    b. Change the working directory to where the wu_py379.yml file is using **cd** in terminal (e.g. "cd Desktop" or "cd Downloads")
    c. Type in the following and press Return/Enter: "conda env create -f wu_py379.yml"
        i. This should prompt Anaconda to install all necessary packages to get you started on the tutorials and analysis
    d. You can check if your virtual environment has been installed by typing in "conda env list" in the anaconda prompt and it should list the name of the virtual environment (wu_py379).

Adding Virtual Environment to Jupyter Notebook Kernels

1. Open Anaconda Prompt (if you haven't done so)
   a. For Windows, click Start and search for "Anaconda Prompt" from the menu and select it
   b. For macOS, press Command and Space simultaneously to open Spotlight and type Terminal. Press Return/Enter
2. Activate your Virtual Environment
   a. Type "conda activate wu_py379" and press Return/Enter button
3. Add it to Jupyter Notebook
   a. Type "python -m ipykernel install --user --name=wu_py379" and press Return/Enter.
4. Deactivate virtual environment
   a. Type "conda deactivate" and press Return/Enter
5. You can exit out of the terminal

To use your virtual environment on Jupyter Notebook, you need to select the kernel you use for the Notebook. Open Jupyter Notebook as usual and select wu_py379 when opening any of the tutorials/analysis notebooks (see below).

Alternatively, you can choose what kernel in the Notebook, and the kernel should be visible in the top right (crudely drawn red circle)



1 Electrocardiogram (ECG)