Environment Setup

DS-GA 1007

Objectives

By the end of this lab, you should have the following installed:

- Anaconda / Miniconda
- Python 3.6+
- NumPy / SciPy / pandas
- Jupyter Notebooks

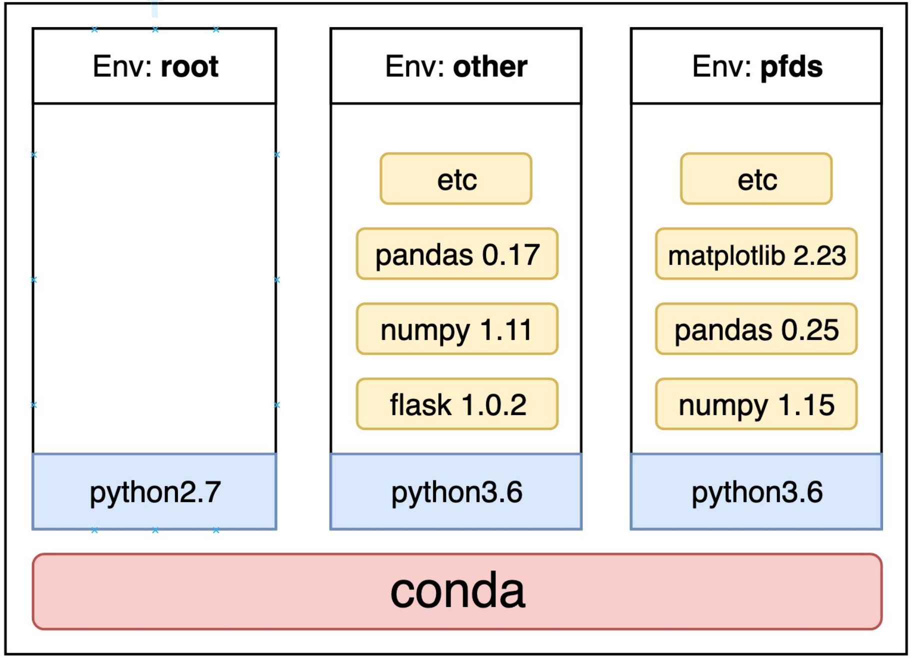
Heads-up

- Linux / MacOS users: You'll be fine, try to make your setup as clean and minimal as possible.

- Windows users: You'll likely run into issues. Ask for help on the forums - other Windows users will likely have similar problems.

If you need help, let us know! Try not to leave this lab without a fully working setup.

Setup Overview



Env: **root**

python2.7

Env: **other**

etc

pandas 0.17

numpy 1.11

flask 1.0.2

python3.6

Env: **pfds**

etc

matplotlib 2.23

pandas 0.25

numpy 1.15

python3.6

conda

Conda Installation

For people starting fresh (never installed anaconda/miniconda, or "what is conda").

- conda is a package and environment manager for Python, which allows you to easily manage multiple environments and has prepackaged libraries/dependencies.
- **Everyone**:
  - Download corresponding installer from: https://conda.io/miniconda.html (Python 3.7)
- **MacOS / Linux users**:
  - Run `bash [installer_file_that_ends_with.sh]`
  - Go with default arguments, let the setup script modify your `.bashrc`
  - MacOS users may need to first install XCode.
- **Windows users**:
  - Run the installer executable
  - I recommend choosing to modify the PATH variable (easier system-wide usage of Conda), but you can also go with the default.
- run `conda list` after to confirm that installation succeeded

For more details: https://conda.io/docs/user-guide/install/index.html

Conda Environment Setup

- Open the respective terminal for your OS:
  - Terminal (Linux/MacOS), Command prompt (Windows) or Anaconda prompt (Windows without modifying path)
- Run `conda create -n pfds python=3.7`
  - This creates a standalone conda environment for this class
  - *Try to never install anything in the root environment*
- Run `conda activate pfds`
  - This activates the environment for this class
- Run `conda install jupyter notebook matplotlib scikit-learn`

Useful Conda commands

- `conda install [package-name]`
  - Install package(s) into current environment

- `conda list`
  - Shows packages installed in current environment

- `conda info --envs`
  - Shows conda environments on your system

Jupyter

- **Demonstration**

- Optional: Run Jupyter persistently in a tmux session! (MacOS/Linux only)
  - Run `tmux new -s pfds` (opens a side session)
  - Activate environment, open notebook, etc
  - Press "Ctrl+B" and then "D" (your notebook session is still running, you can even close the console)
  - Run `tmux attach -t pfds` (reopens the `pfds` tmux session)

System / Python Setup

- To import packages in Python, the root folder of the package needs to be on your PYTHONPATH.
- You can see it in `import sys; print(sys.path)`
- The current working directory is always included in the PYTHONPATH.
- There are several common ways to modify your PYTHONPATH.
  1. Modify at the system level
     - e.g. Adding `export PYTHONPATH=/my/new/path:$PYTHONPATH` to `.bashrc`
  2. Modify on the fly
     - Running `export PYTHONPATH=/my/new/path:$PYTHONPATH` before running your code / starting your notebook server.
  3. Modify for just that command
     - Running `PYTHONPATH=/my/new/path:$PYTHONPATH python`
  4. Modify in-session:
     - In a Python session, run:

     -
       ```
       import sys
       sys.path += ["/my/new/path"]
       ```

Editors

- **Sublime**: Free, lightweight and excellent text editor

- **PyCharm**: Full-fledged IDE
  - Community edition available for free
  - Professional licence available for free to students
  - Learn to set up *projects*

- **VS Code**
  - More language agnostic

Git

- Git is a version control system
  - GitHub is a website that's built on Git

- Install Git to your machine following instructions on https://git-scm.com/book/en/v2/Getting-Started-Installing-Git