Introduction to Jupyter
SHARCNET General Interest Webinar

preney@sharcnet.ca

School of Computer Science
University of Windsor
Windsor, Ontario, Canada

Copyright © 2016 Paul Preney. All Rights Reserved.

August 17, 2016
1. Overview

2. Installation

3. Jupyter Notebook
Overview
- What is Project Jupyter?
- What is IPython?
- Jupyter Project: Notebook
- Jupyter Project: nbviewer
- Jupyter Project: tmpnb
- Jupyter Project: Jupyter Hub
What is Project Jupyter?

- Started in 2014.
- It is a spin-off of IPython.
  - IPython continues to exist as a shell and kernel for Project Jupyter.
  - IPython Notebook and other components of IPython have been moved to Jupyter.
  - Project Jupyter’s tools’ primary mode of interaction is via the web —not the command line.
What is IPython?

- Released in 2001.
- IPython is a command line shell which enables interactive computing using Python and many other programming languages.
- The IPython Notebook feature provides interaction via programming languages and data visualization.
  - IPython Notebook is now Jupyter Notebook.
- Can be used for data science.
- Can be used as an interactive front-end to parallel computing technologies and applications.
Jupyter Notebook:

- is the most commonly used Jupyter Project
- is a web application
  - i.e., you use a web browser to access it
- enables the creation of interactive, sharable documents with live code, visualizations, and text

NOTE: You can also still use IPython Notebook to access notebooks as well if using a web browser is not feasible.
Uses include **interactive**:

- data science
- statistical modelling
- big data
- machine learning
- data cleaning and transformation
- interfacing with parallel back-ends (e.g., Hadoop, Spark)
- interchanging data across programming languages
Jupyter nbviewer:

- is a web application
- enables non-live, read-only Jupyter Notebook document sharing via the web, e.g.,
  - to share published research work done with IPython notebooks
- e.g., http://nbviewer.jupyter.org which has many example notebooks that can also be downloaded and used

**NOTE:** System and/or web administrators would set this up.
Jupyter tmpnb:

- is short for **Temporary Notebook**
- is a web application
- enables live (interactive), ephemeral Jupyter Notebook document viewing and sharing via the web
  - Jupyter nbviewer is read-only —this is live.
  - Changes made to notebooks are not saved and will be reset.

- e.g., `https://try.jupyter.org` is a Jupyter Notebook demonstration web site. Do check it out!

**NOTE:** System and/or web administrators would set this up.
Jupyter Project: Jupyter Hub

Jupyter Hub:

- is a multi-user web application
- enables live (interactive), Jupyter Notebook authenticated multi-user document creation, editing, and sharing via the web
  - Jupyter tmpnb doesn’t save edits —this does.
- Excellent for research / group projects.
- All files are stored on resources connected to the Jupyter Hub web server.
- HPC and advanced computing sites running Jupyter Hub would make available personal (non-shared) and team (shared) home and data directories to make it easy for everyone to work with them.

**NOTE:** System and/or web administrators would set this up.
Installation
- Miniconda
- Installing Miniconda
- Configuring Conda Environments
- Activating a Conda Environment
- Deactivating a Conda Environment
The easiest way to install IPython and Project Jupyter software for single-user use is to install **Miniconda**:  
- Supports **separate** Python environments.  
  - e.g., Python 2 with legacy tools, Python 3, Python 3 with different versions of tools installed.
- Switching between these environments is straight-forward.

**NOTE:** You will need a command-line terminal window to install Miniconda as well as to start and stop Jupyter Notebook.
Installing Miniconda

To install Miniconda:

- Go to http://conda.pydata.org/miniconda.html.
- Download and the appropriate installer for your operating system.
  - Miniconda will install the necessary tools (including Python).
- Run the installer.
  - e.g. on Linux: sh Miniconda3-latest-Linux-x86_64.sh
- On Linux and Mac machines, it will ask if you want to modify ~/.bashrc.
  - If you say yes, then you will need to run source ~/.bashrc to use Conda in that window.
  - If you say no, then you will need to save the line it outputs so you can run it later to use Conda.
Open a new terminal (command-line) window.

If you did not modify your .bashrc, then run:

- `export PATH=~/.miniconda3/bin:$PATH` assuming you installed Miniconda to the default location.

Create a Python 2 environment. From the command line install Anaconda w/Python version 2:

- `conda create -n py2 python=2 anaconda`

Create a Python 3 environment. From the command line install Anaconda w/Python version 3:

- `conda create -n py3 python=3 anaconda`
Activating a Conda Environment

Before running Jupyter Notebook you need to activate the desired Conda environment:

- To activate the py2 Conda environment:
  - `source activate py2`

- To activate the py3 Conda environment:
  - `source activate py3`

**NOTE:** A Conda environment is only active in the terminal window it has been activated in.
Deactivating a Conda Environment

To deactivate (unload) the current conda environment:

- `source deactivate`
Table of Contents

3 Jupyter Notebook
- Running Jupyter Notebook
- Some Useful Libraries
Running Jupyter Notebook

- To start Jupyter Notebook Web Server:
  - Activate the desired Conda environment.
  - Run: jupyter notebook

- To stop the Jupyter Notebook Web Server:
  - Hit Ctrl-C twice.
Some Useful Libraries

- **matplotlib (2D Plotting Library)**
  - http://matplotlib.org

- **Pandas (Data Analysis Library)**
  - http://pandas.pydata.org

- **SciKit-Learn (Machine Learning)**
  - http://scikit-learn.org

- **Seaborn (Statistical Data Visualization)**
  - http://stanford.edu/~mwaskom/software/seaborn/

- **StatsModel (Statistical Data Analysis Library)**
  - http://statsmodels.sourceforge.net/