Jupyter Notebook

Introduction

To set up using conda…

conda create -n my_notebook python=3.5
activate my_notebook
conda install jupyter notebook nb_conda_kernels
mkdir my_notebook
cd my_notebook
jupyter notebook

A web browser will open and you can navigate to your “my_notebook” conda environment. Once there you can type in code and mix it with markdown. The entire thing seems a bit hookie.

Shortcuts

A list of keyboard shortcuts can be shown by pressing H
Press Ctrl+Enter to compile and run the code/markdown
Press Shift+Enter to compile and run the code/markdown and move to the next cell

Magic keywords

Magic keywords are special commands you can run in cells that let you control the notebook itself or perform system calls such as changing directories. For example, you can set up matplotlib to work interactively with notebook with %matplotlib.

Magic commands are preceeded with one or two % signs (% = line magics, %% = cell magics). Line magics apply to the line while cell magics apply to the entire cell.

Timing Code

You can use %timeit <code here> to automatically have the notebook time how long it takes the code to run. It'll be shown in the output. If you want to see how long it takes for a whole cell to run, use %%timeit at the beginning.

Visualizations

You can use %matplotlib inline to embed your matplotlib plots into your notebook. On higher res screens the generated images may be blurry, so put %config InlineBackend.figure_format = 'retina' after %matplotlib inline to render highres plots.

Debugging

Use %pdb so you can inspect variables when you have issues with your code breaking. It gives you an interactive pdb prompt.

Converting Notebooks to HTML/Slideshows

jupyter nbconvert --to html notebook.ipynb

jupyter nbconvert --to slides notebook.ipynb --post serve
Taking out the --post serve part will create the files but won't open the browser. It seems like to create slideshows you need to designate the relevant cells as part of the slide.