

Jupyter tutorial

August 28, 2019

Course: *Modélisation mathématique et computationnelle en biologie* (BIO-341)

Professor: *Felix Naef*

SSV, BA5, 2019

Adapted from the [Jupyter tutorial](#) by [Allison Parrish](#).

1 Jupyter Notebook Tutorial

Jupyter Notebook gives you a convenient way to experiment with Python code, interspersing your experiments with notes and documentation. You can do all of this without having to muck about on the command line, and the resulting file can be easily published and shared with other people. In this course, I'll be using Jupyter Notebook to do in-class examples, and the notes will be made available as Jupyter Notebooks. Some of the homeworks will be assigned in the form of Jupyter Notebooks as well.

An Jupyter Notebook consists of a number of "cells," stacked on the page from top to bottom. Cells can have text or code in them. You can change a cell's type using the "Cell" menu at the top of the page; go to `Cell > Cell Type` and select either `Code` for Python code or `Markdown` for text. (You can also change this for the current cell using the drop-down menu in the toolbar.)

1.1 Text cells

Make a new cell, change its type to `Markdown`, type some stuff and press `Ctrl-Enter`. Jupyter Notebook will "render" the text and display it on the page in rendered format. You can hit `Enter` or click in the cell to edit its contents again. Text in `Markdown` cells is rendered according to a set of conventions called `Markdown`. `Markdown` is a simple language for marking up text with basic text formatting information (such as bold, italics, hyperlinks, tables, etc.). [Here's a tutorial](#). You'll also be learning `Markdown` in more detail in the Foundations course.

1.2 Code cells

You can also press `Alt-Enter` to render the current cell and create a new cell. New cells will by default be `Code` cells. Try it now!

```
In [1]: print("This is a code cell.")
        print("")
        print("Any Python code you type in this cell will be run when you press the 'Run' button")
        print("or when you press Ctrl-Enter.")
        print("")
```

```
print("If the code evaluates to something, or if it produces output, that output will be  
shown beneath the cell after you run it.")
```

This is a code cell.

Any Python code you type in this cell will be run when you press the 'Run' button, or when you press Ctrl-Enter.

If the code evaluates to something, or if it produces output, that output will be shown beneath the cell after you run it.

```
In [2]: print("If your Python code generates an error, the error will be displayed in addition  
        print("to any output already produced.")
```

```
1 / 0
```

If your Python code generates an error, the error will be displayed in addition to any output already produced.

```
-----  
  
ZeroDivisionError                                Traceback (most recent call last)  
  
  <ipython-input-2-b1c1f52e3cdc> in <module>()  
    2 print("to any output already produced.")  
    3  
----> 4 1 / 0  
  
ZeroDivisionError: division by zero
```

Any variables you define or modules you import in one code cell will be available in subsequent code cells. Start with this:

```
In [3]: import random  
        stuff = ["cheddar", "daguerrotype", "elephant", "flea market"]
```

... and in subsequent cells you can do this:

```
In [4]: print(random.choice(stuff))
```

elephant

1.3 Keyboard shortcuts

As mentioned above, `Ctrl-Enter` runs the current cell; `Alt-Enter` runs the current cell and then creates a new cell. `Enter` will start editing whichever cell is currently selected. To quit editing a cell, hit `Esc`. If the cursor isn't currently active in any cell (i.e., after you've hit `Esc`), a number of other keyboard shortcuts are available to you:

- `m` converts the selected cell to a Markdown cell
- `b` inserts a new cell below the selected one
- `x` "cuts" the selected cell; `v` pastes a previously cut cell below the selected cell
- `h` brings up a help screen with many more shortcuts.

1.4 Saving your work

Hit `Cmd-S` at any time to save your notebook. Jupyter Notebook also automatically saves occasionally. Make sure to give your notebook a descriptive title by clicking on "Untitled0" at the top of the page and replacing the text accordingly. Notebooks you save will be available on your server whenever you log in again, from wherever you log into the server.

You can "download" your notebook in various formats via `File > Download as`. You can download your notebook as a static HTML file (for, e.g., uploading to a web site), or as a `.ipynb` file, which you can share with other people who have Jupyter Notebook or make available online through, e.g., [nbviewer](#).