**Jupyter Notebooks**

We will start with a very very short introduction to Jupyter Notebooks – just what we need to work through this workshop.

Jupyter is a project to develop open-source software. In many different languages. And the most useful tool is the Jupyter Notebooks, that is a web application to create interactive scripts. Most of the time, you’ll be running in your computer, but still it is accessed through your web browser. You could also run them on a remote server (as we are doing).

But Jupyter Notebook is more than just an interactive script interface. It lets you enter and save text visualizations, equations, and save them, so you can see them later without needed to rerun your code. Here an example: text include formatting and links, these cells are code cells, graphics will show every time you open the file, and you could add and visualize equations.

We will cover the basics, but more detailed tutorials and useful links to installation and use are provided at the end of this notebook.

You were provided for instructions for installing python and jupyter notebooks, but if you have questions about it, we can talk about it at the end of the workshop.

From this moment on, the workshop will be practical, and all text in blue boxes are instructions for you to try our.

Let’s start with the dashboard. When you launch the Jupyter notebook server, either in your computer or remotely, you’ll see a dashboard like this, where files are listed, and you can create a new file by clicking on the New button at the top right. It will give you choices of programing languages or file type. In our case, we are using python 3.

As a result, we will get a brand new file like this.

**Rename a script**

You can rename a script by clicking on the file space at the top

**Saving your own script**

And you can save your script by clicking this button, or using the control-s key shortcut

For this tutorial, since we are running remotely in mybinder, modifications done in your notebooks and saved to the server will be lost when we close the connections. Therefore if you want to save the modifications you will be doing to the notebooks in your computer you have to go to file, download as, and notebook. Try it out.

**Modes**

Jupyter Notebook has two operation modes. Command and Edit. In command mode, typing and shortcuts applies to the entire script. In Edit mode, it only applies to the selected cell.

If you click on the cell bellow, automatically you enter the Edit mode, which is recognized by the green color on the left of the cell. By pressing Esc, you go into “command” mode, which is shown by the blue color on the right.

The main reason to tell you this is because once we start using shortcuts, sometimes you’ll think your notebook is doing something weird, which is easily explain by the fact that you think your editing, and in reality, you’re in command mode.

**Cell types**

There are two useful cell types, Code and Markdown.

Code cells are executable code, very self explanatory. And markdown cells are cells with formatted text and equations.

You can select the type on the icon menu on the top (center). Or, in command mode, using the letters y and m for code and markdown respectively.

Let’s try it out. What changes with the cell type? Note not only the color of the square but of those of the letters and operation symbols

**To execute commands**

There are two ways to execute code in a cell, with a shortcut or by clicking the run button.

Shortcuts are:

-shift-enter

-control-enter,

The difference is what happens after execution, first case we move or create a cell below and in the second case we stay in the same cell.

You could also execute the entire script, or all cells above by using the cell menu.

**Other commands**

You will need to add, delete and move cells around as you build your code. You could do that from the icon menu or using shortcuts. Let’s try them out

Another very useful shortcut is l, which add the line number to a cell.

Try it out.

What happen if you add it to a markdown cell?

**Kernel**

A last note of warning about the kernel.

The kernel is the little python program or instance running in your or remote computer that is executing or interpreting your code.

Once in a while it dies, or if you get your code stuck in some way you’ll need to restart it.

You can do this using the drop menu above.

(or going to the dashboard, selecting the program you’re working on and click the shutdown menu).

Once you restart, you’ll need to run all the cells above the one you were working on.

Hopefully we won’t use it today, but sooner or later you’ll have to.

And lets start pythoning!