jupyter Notebooks

Adriano Di Florio (INFN & Politecnico Bari)

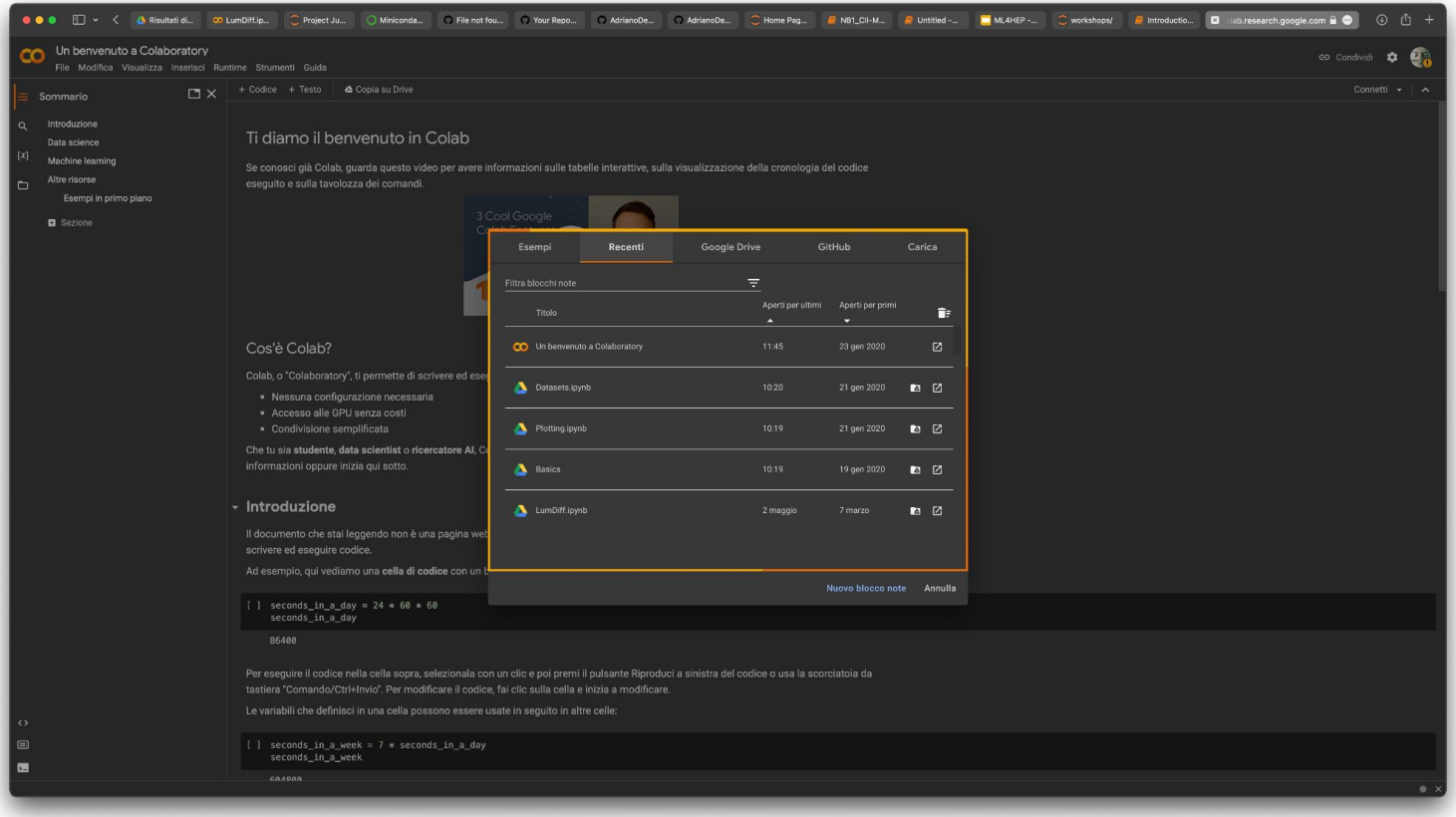
4 May 2023

Datasets

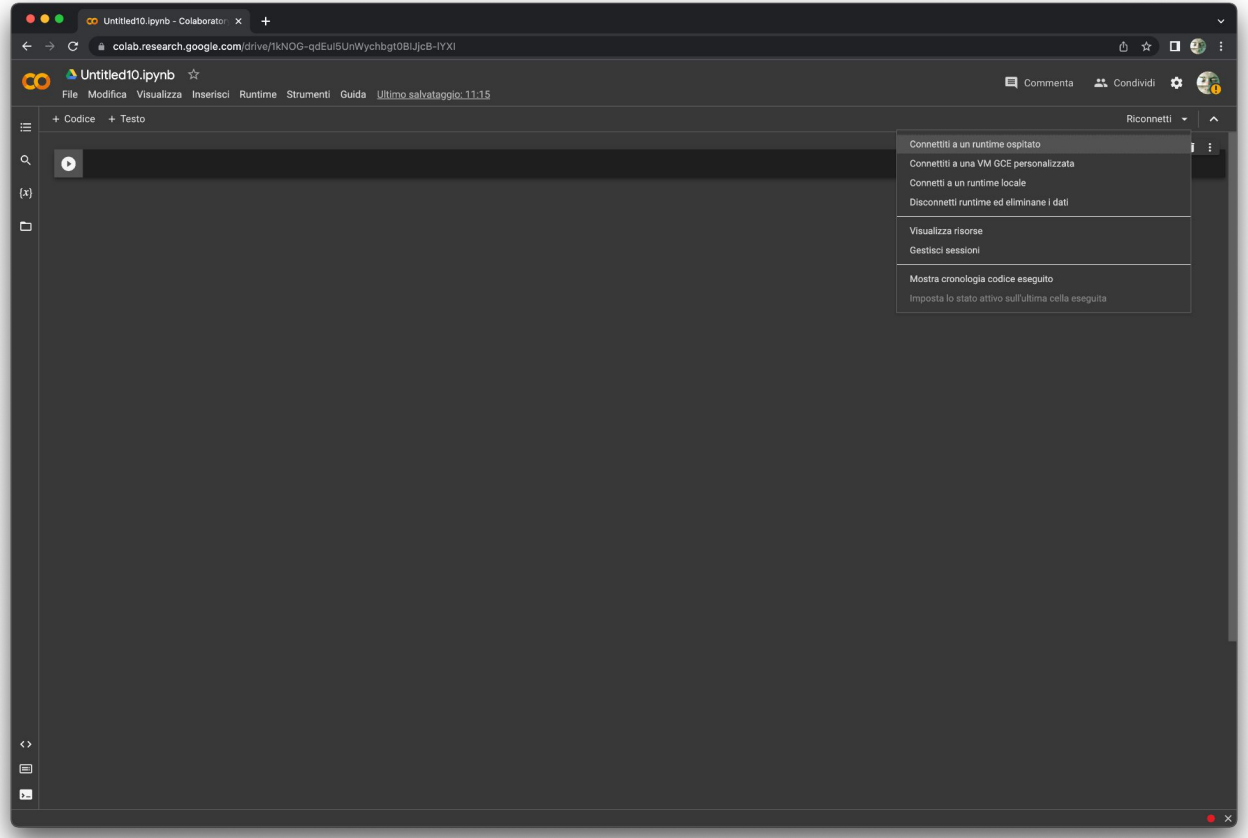
- **Option 1**: using Colab.
- **Option 2**: installing locally.
- **Option 3**: using ReCaS.

Google Colab

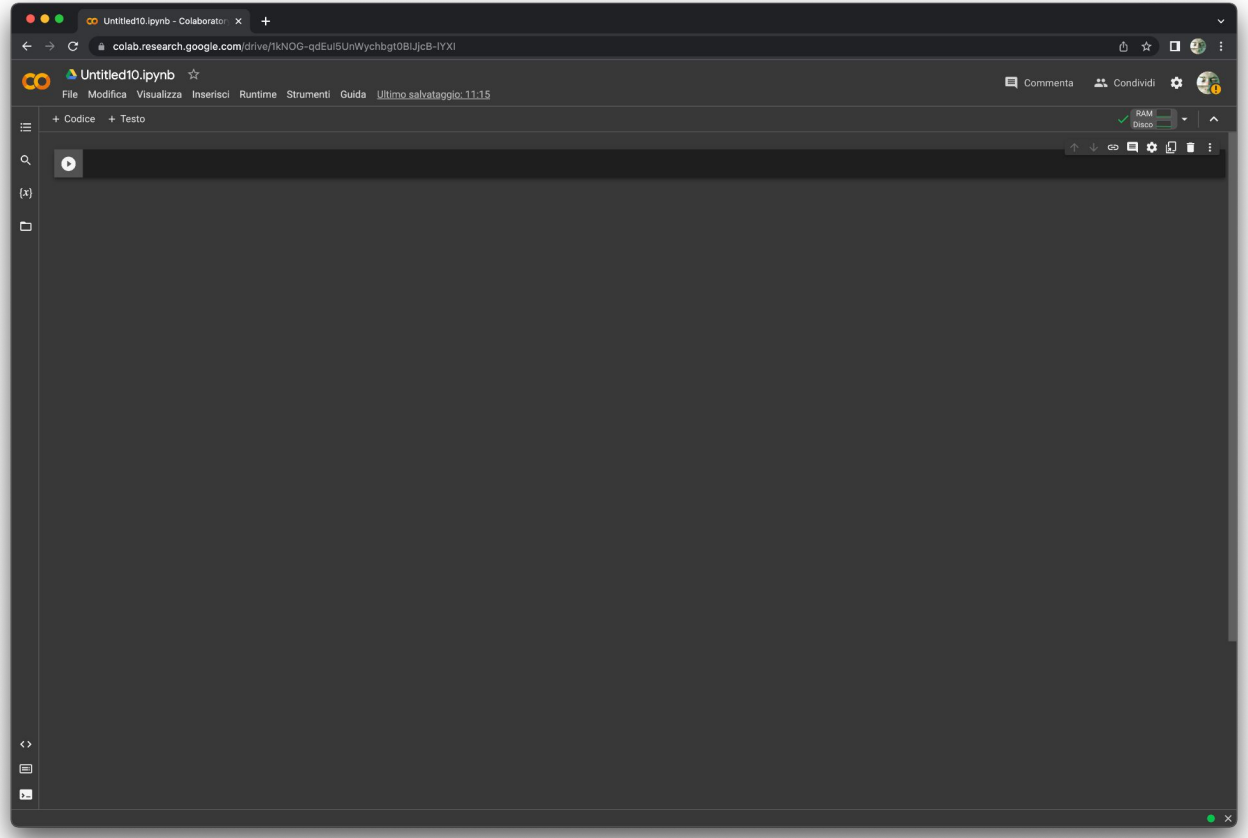
- Go to: http://colab.research.google.com



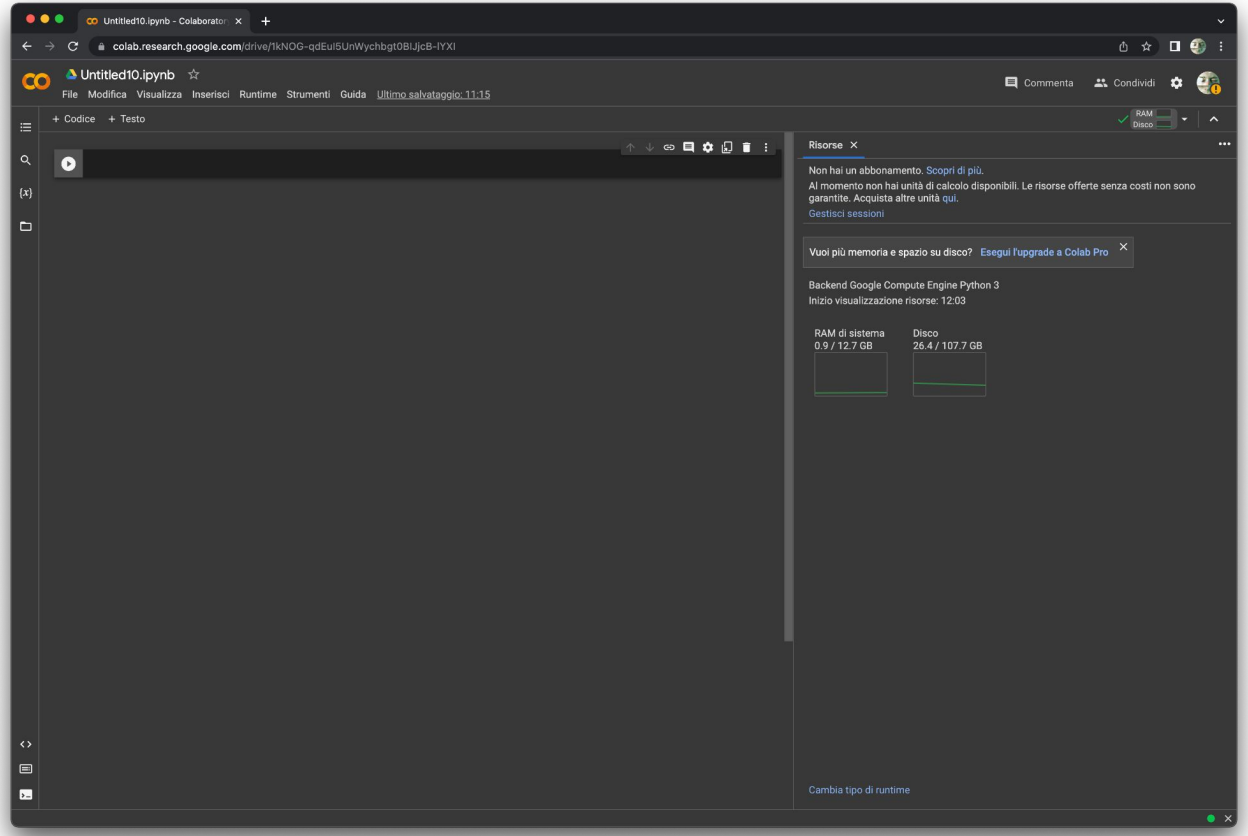
Google Colab



Google Colab

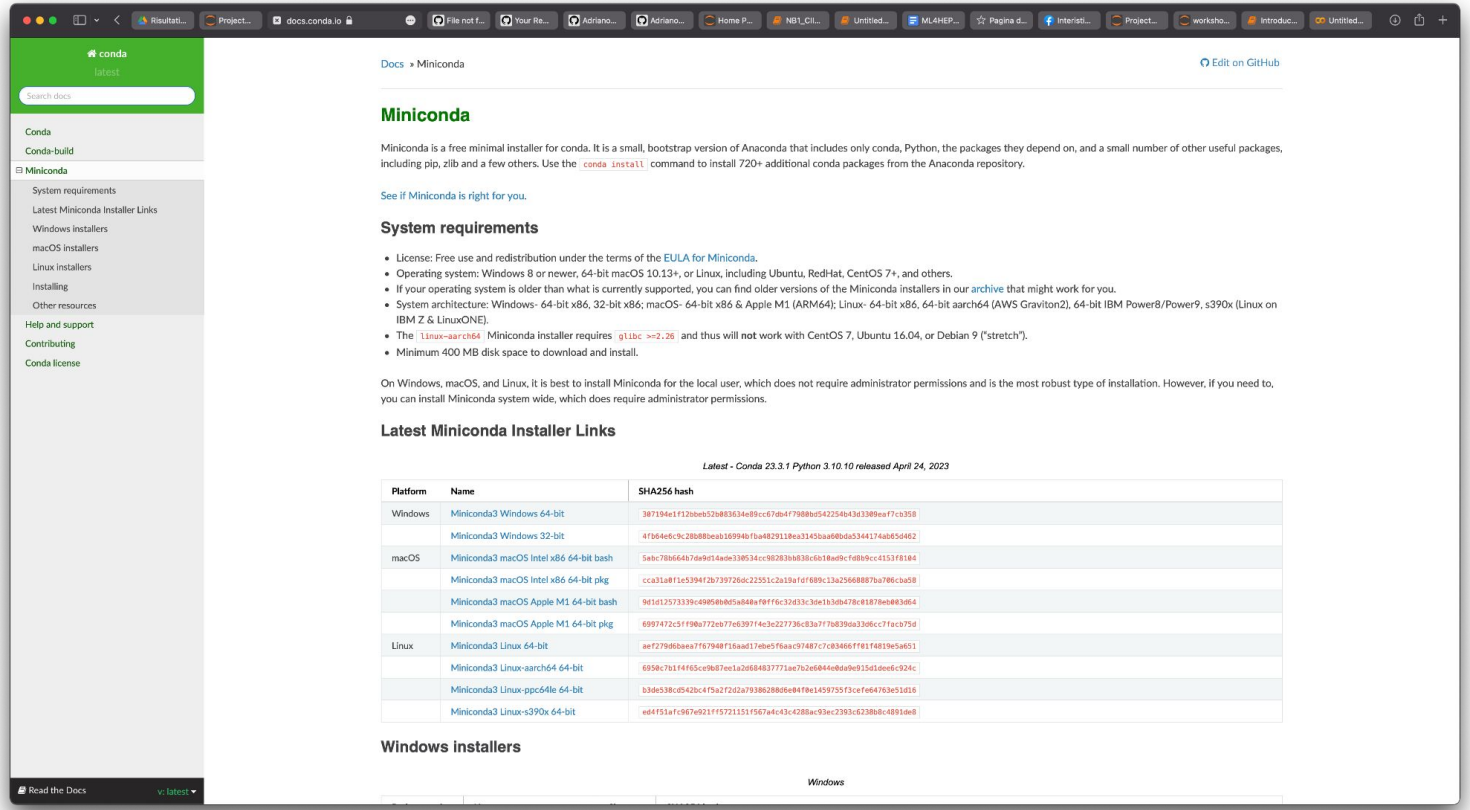


Google Colab



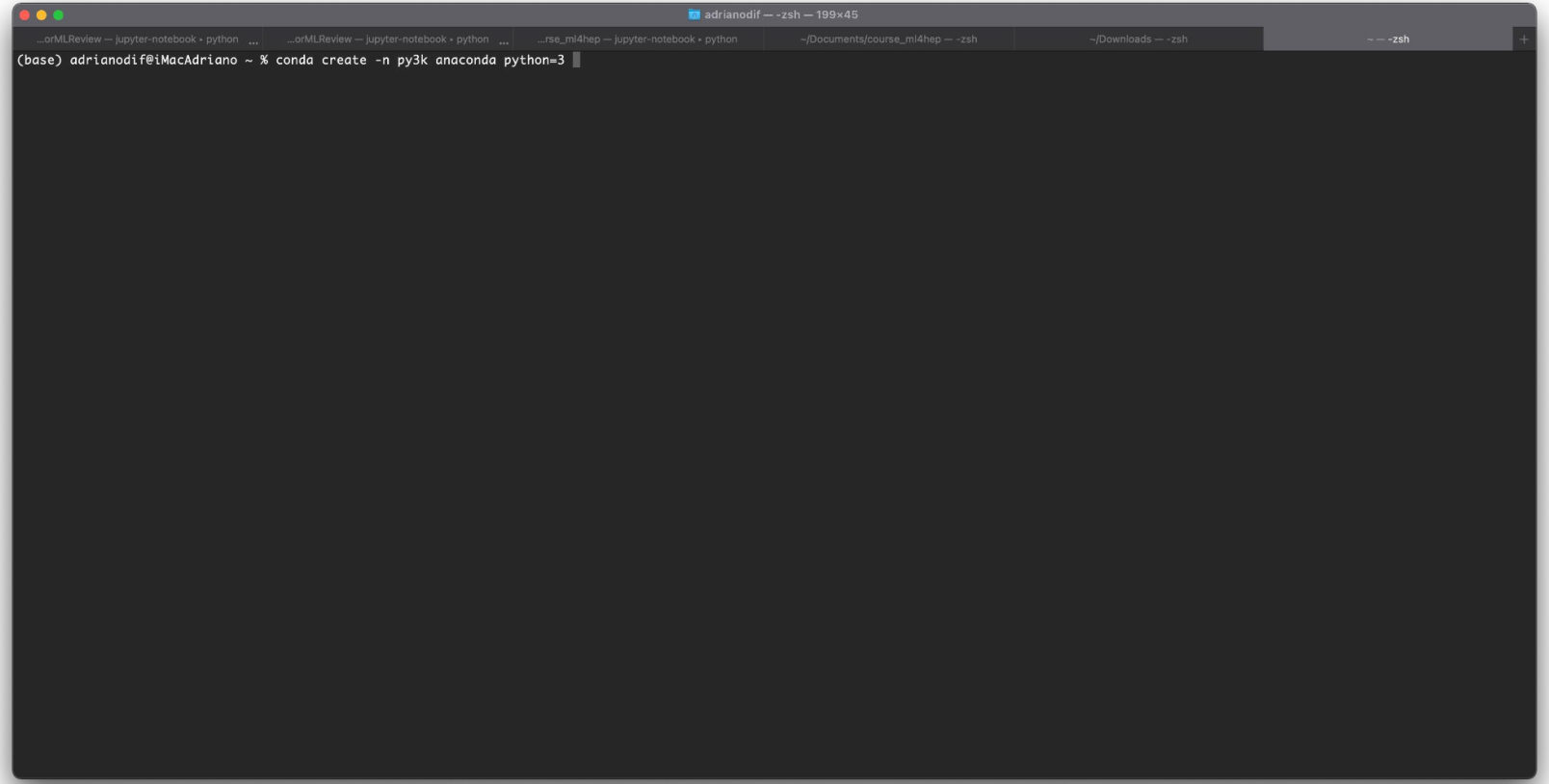
Installing Locally

- Install [MiniConda]



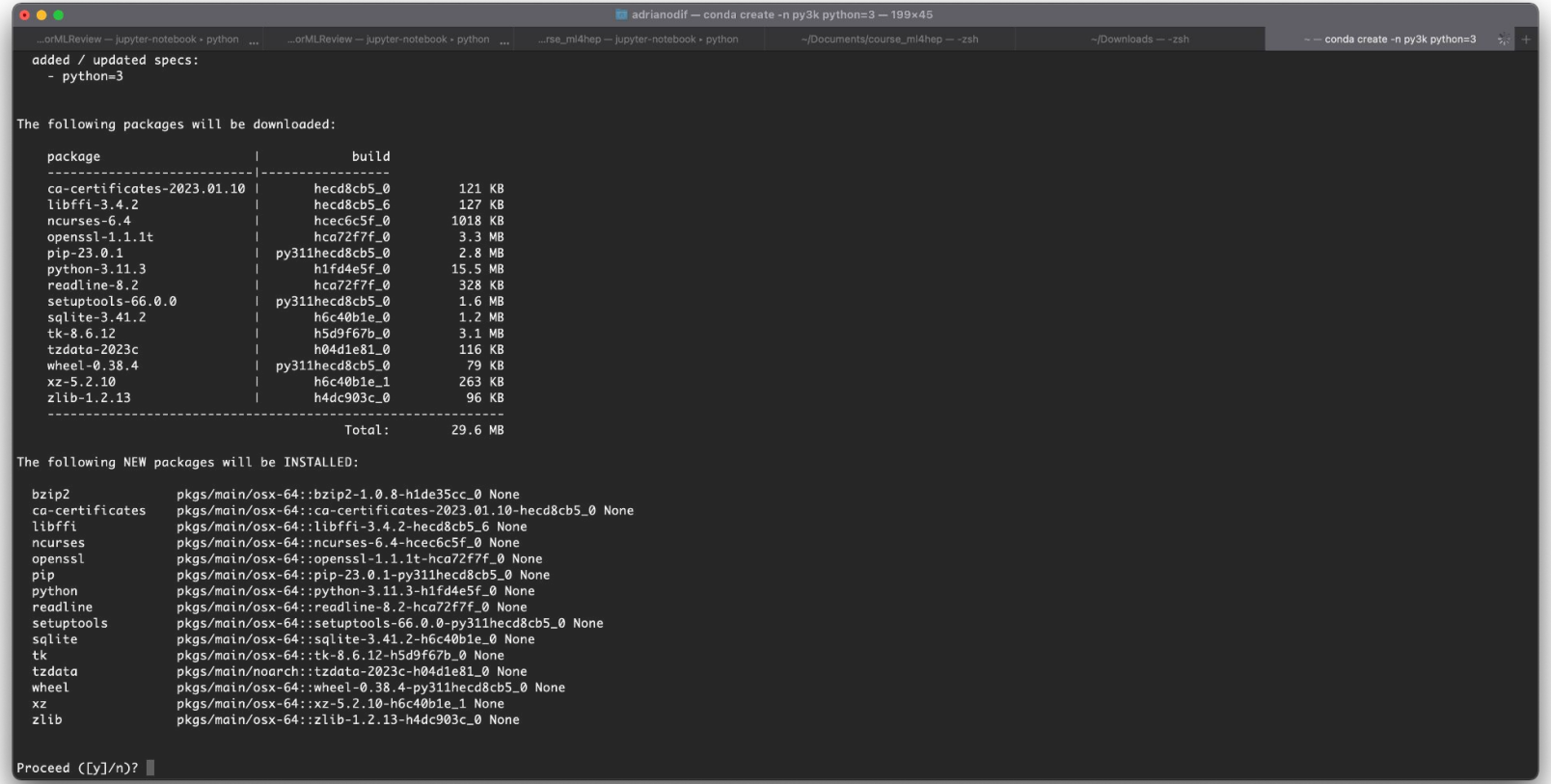
Installing Locally

- Create your **python** environment



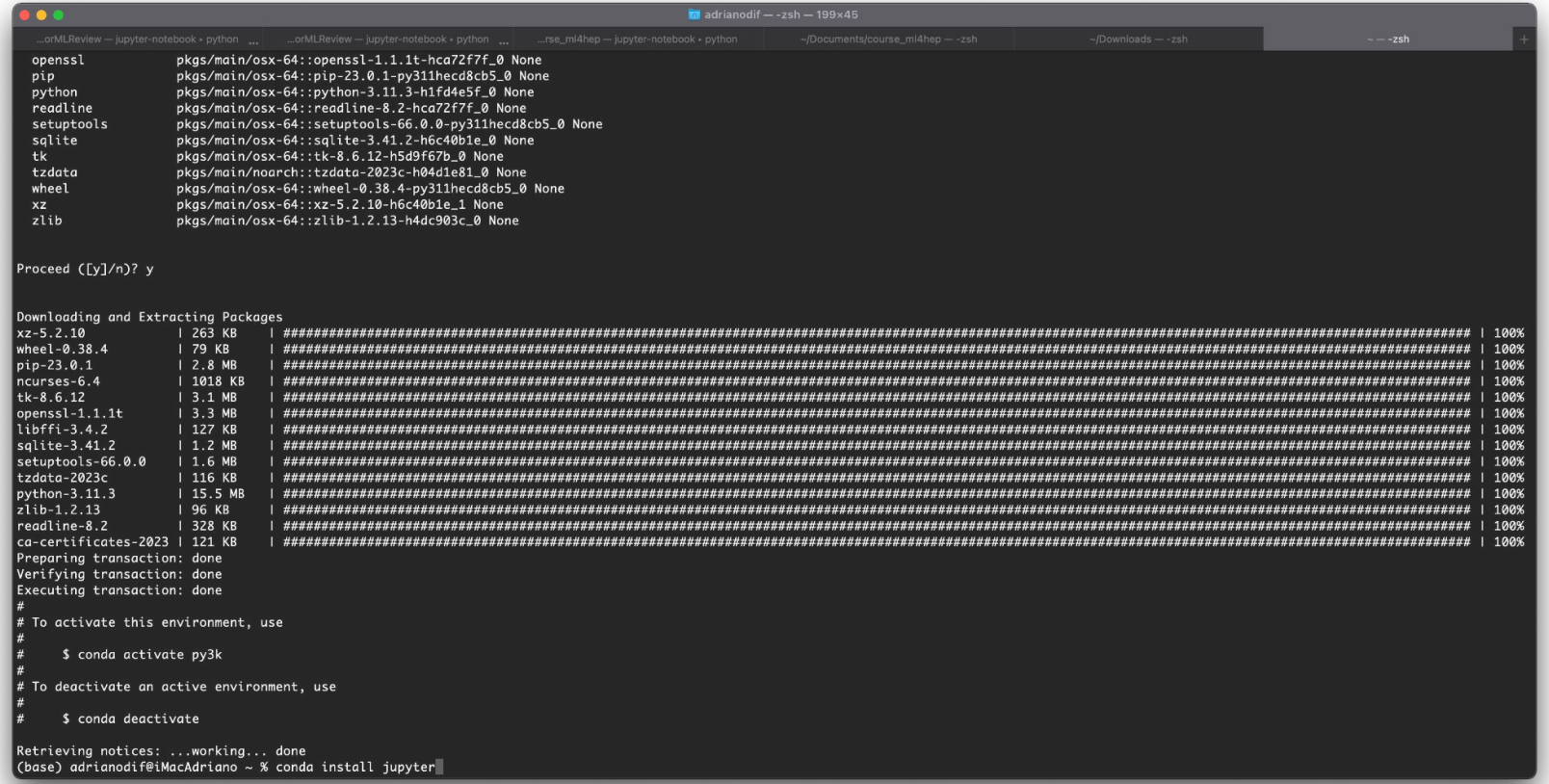
Datasets

- Create your **python** environment



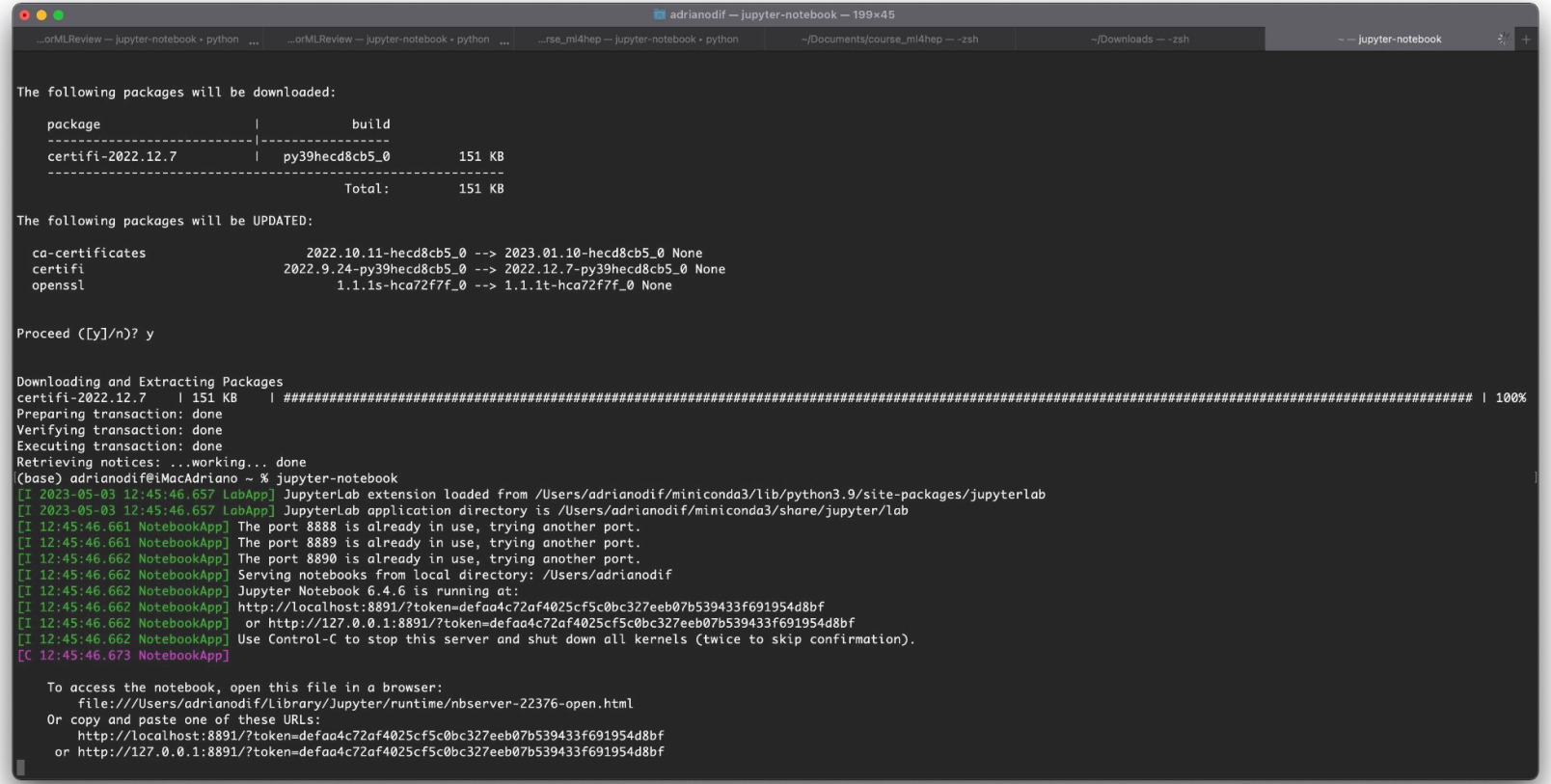
Installing Locally

- Install **jupyter-notebook**



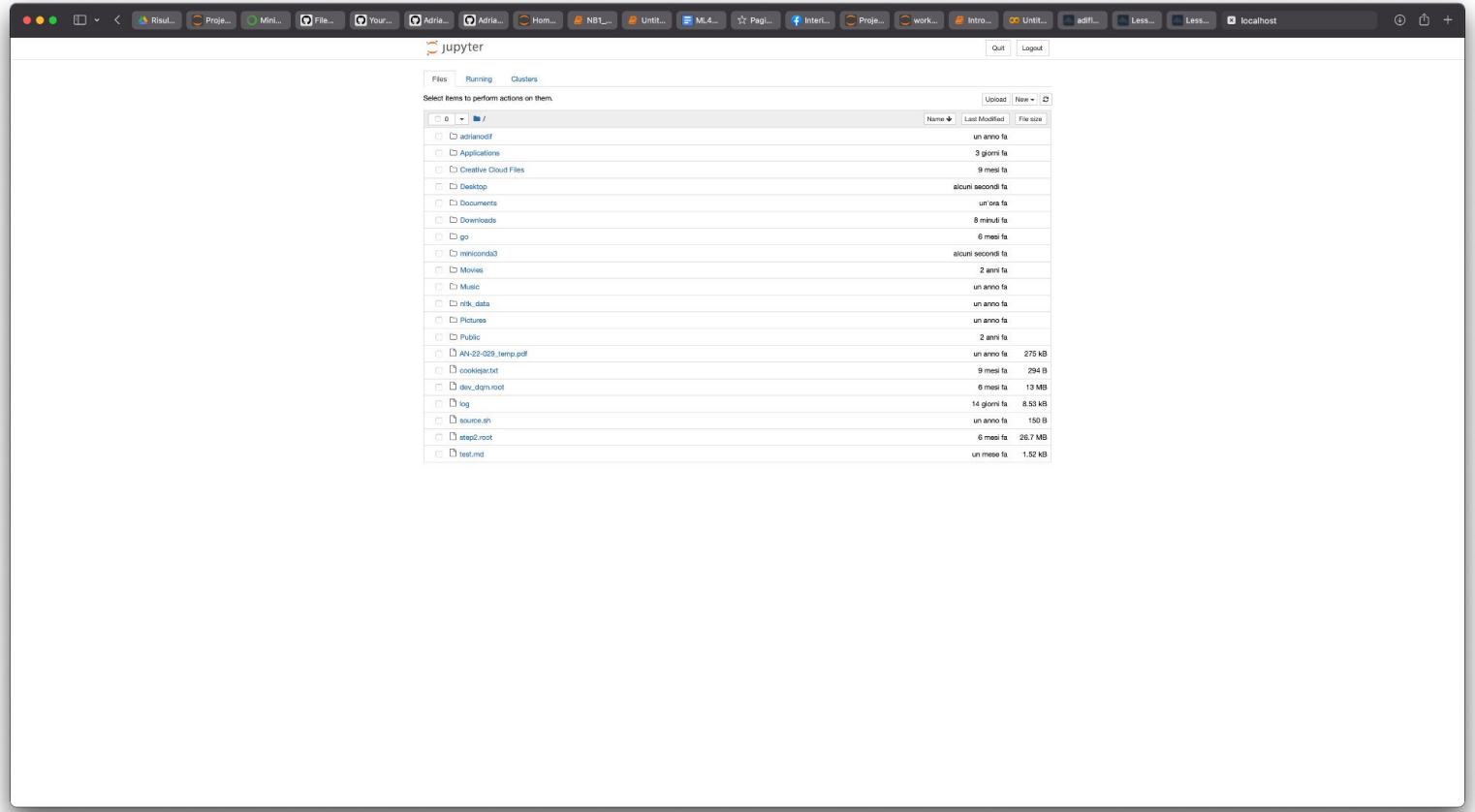
Installing Locally

- Install **jupyter-notebook**



Installing Locally

- Run it!

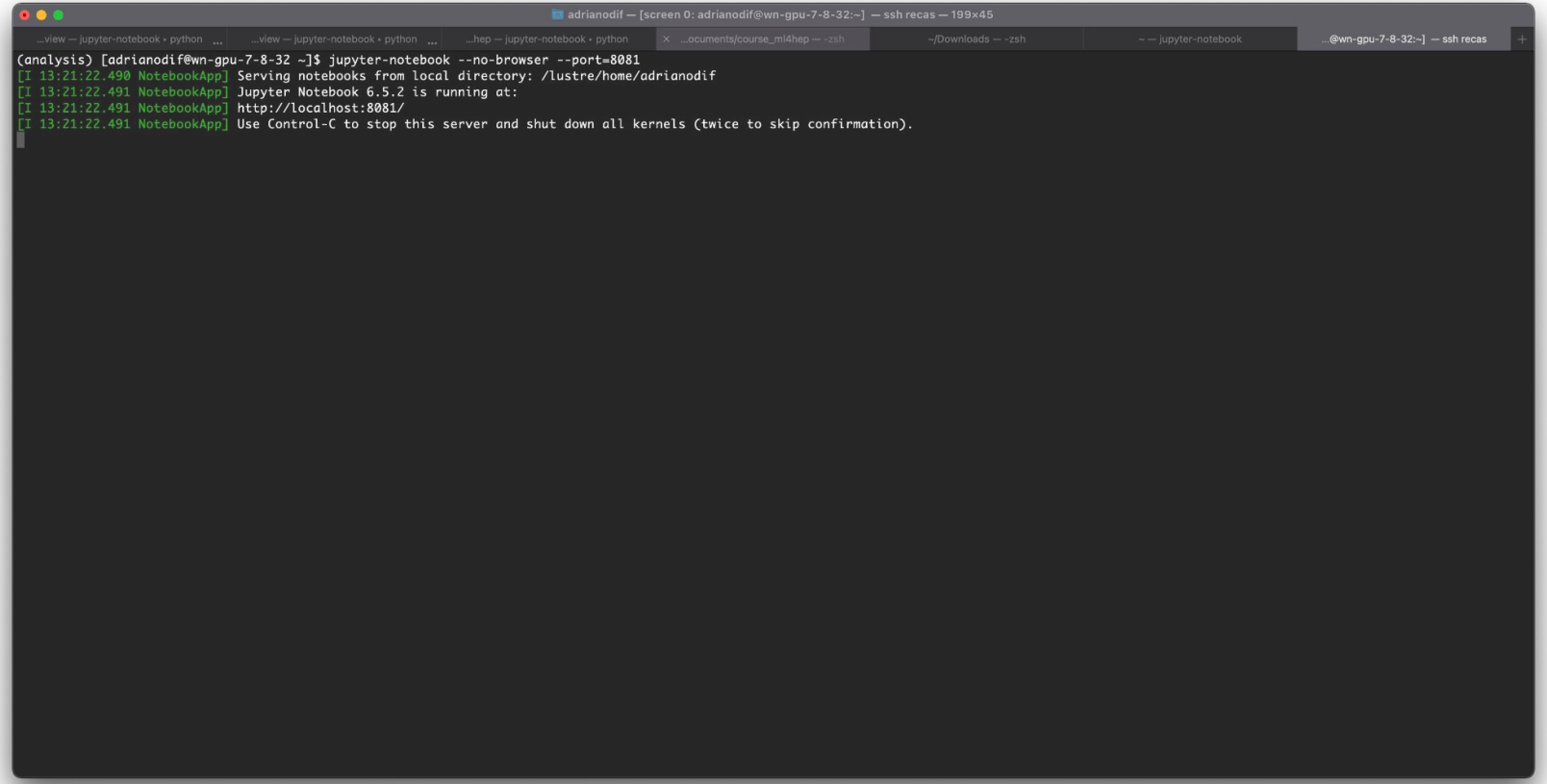


Running on ReCaS

- If you have access to recas HPC/HTC you can run it there having access to the cluster machines.
- First of all follow the instructions given for the local installation on the machine you have access to.
- Then you would need to follow the instructions here to have an **ssh** tunnel to the machine you are logged in and your local machine. There you will also find a summary of the instructions I've sketched above.
- **Warning!** Since there may be other people connected you will need to choose a port that you are ~sure is not already taken (i.e. don't use the common ones 8000 8080 8888 etc.)

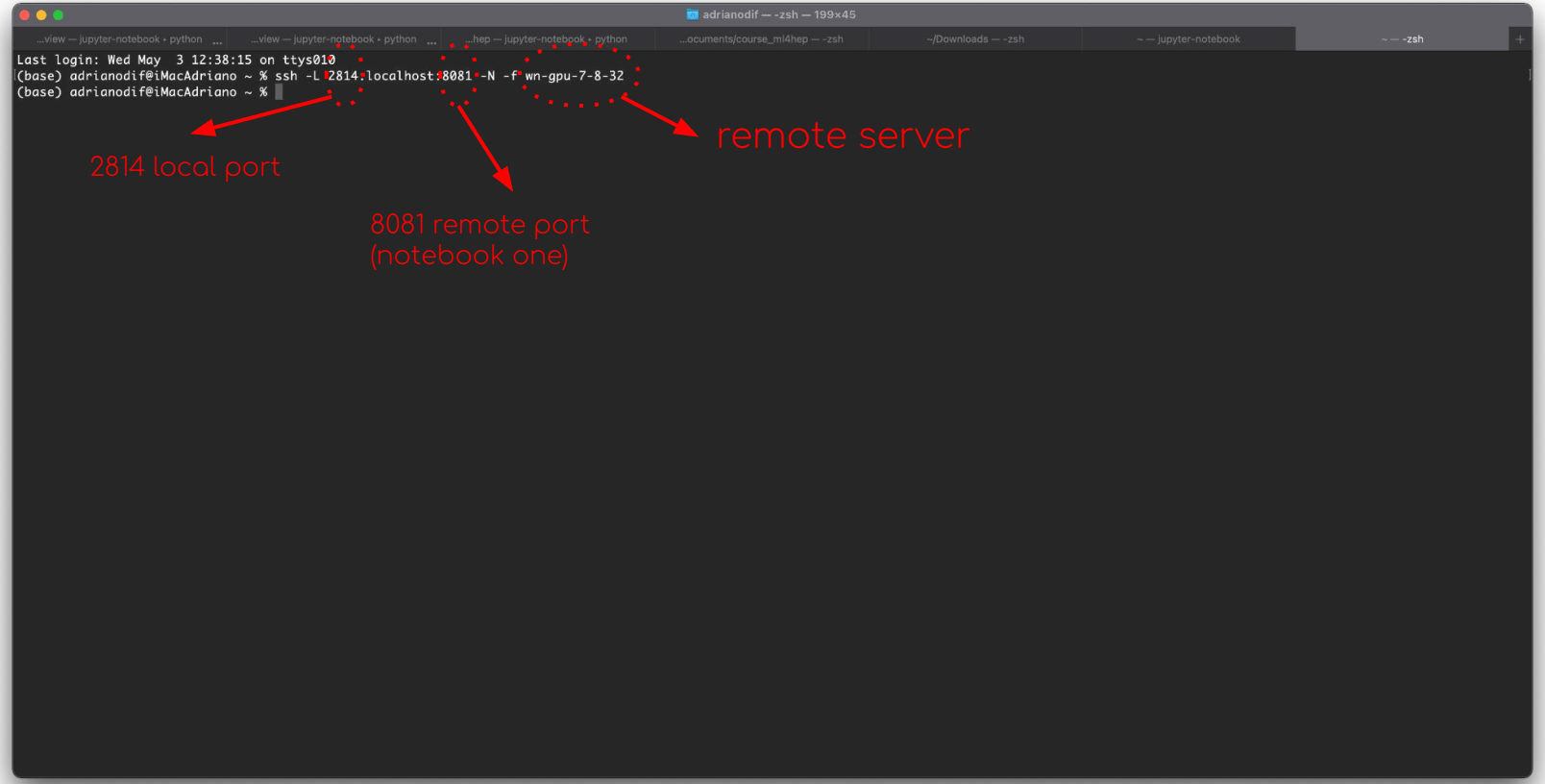
Running on ReCaS

- [on remote/ssh'd machine] **Start** the notebook.



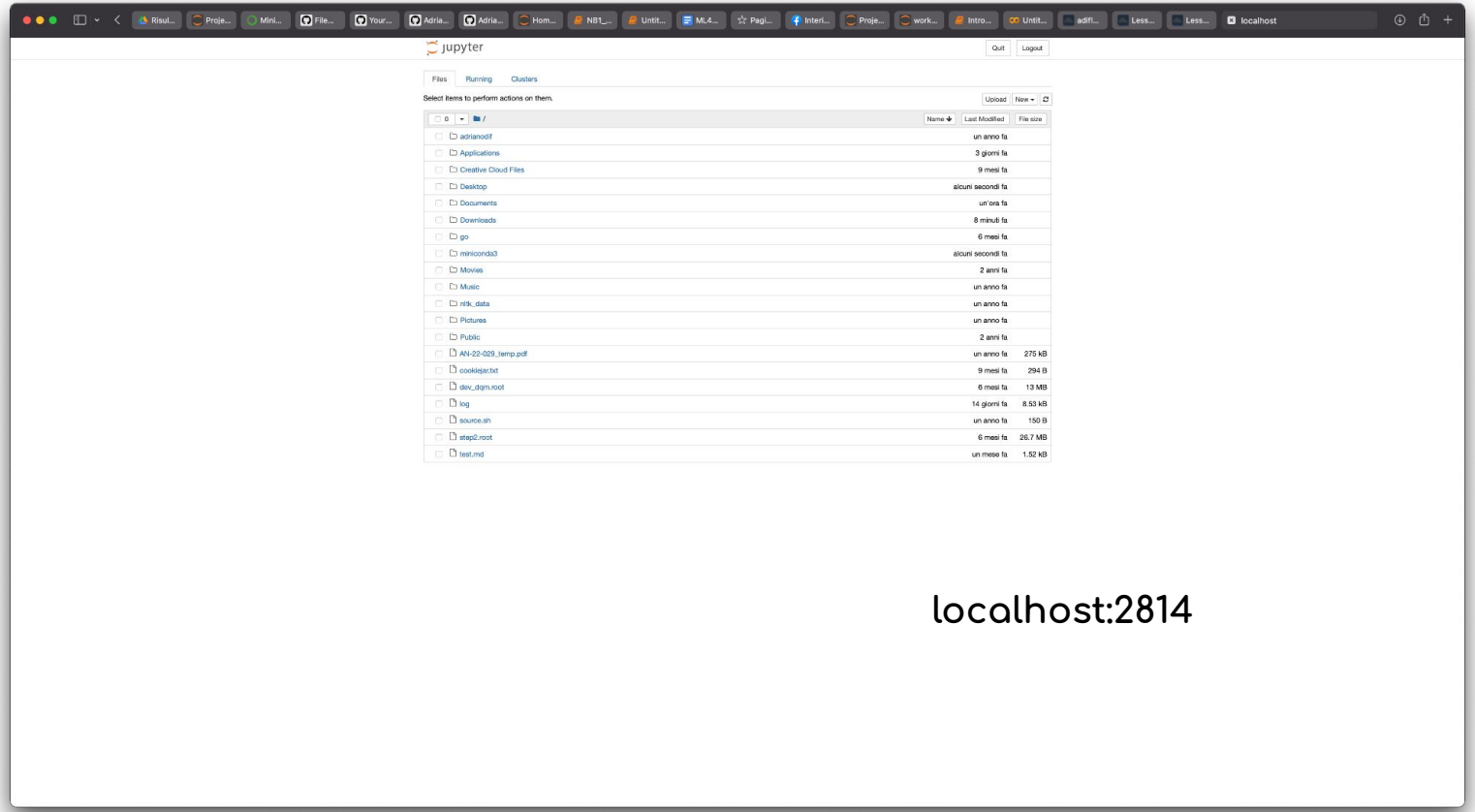
Running on ReCaS

- [on local machine] **Tunneling** from local to remote.



Running on ReCaS

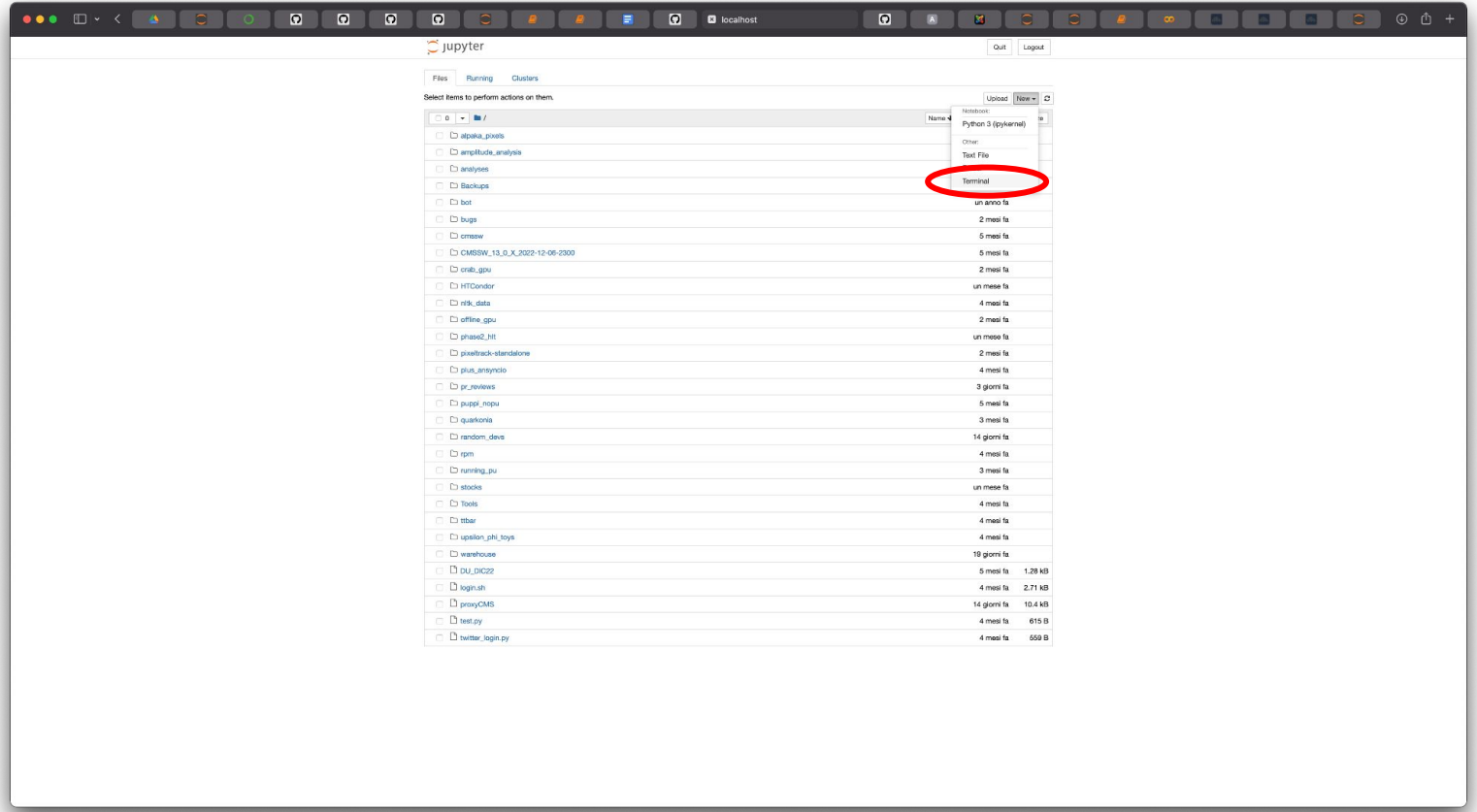
- My notebook running **on a remote machine**.



localhost:2814

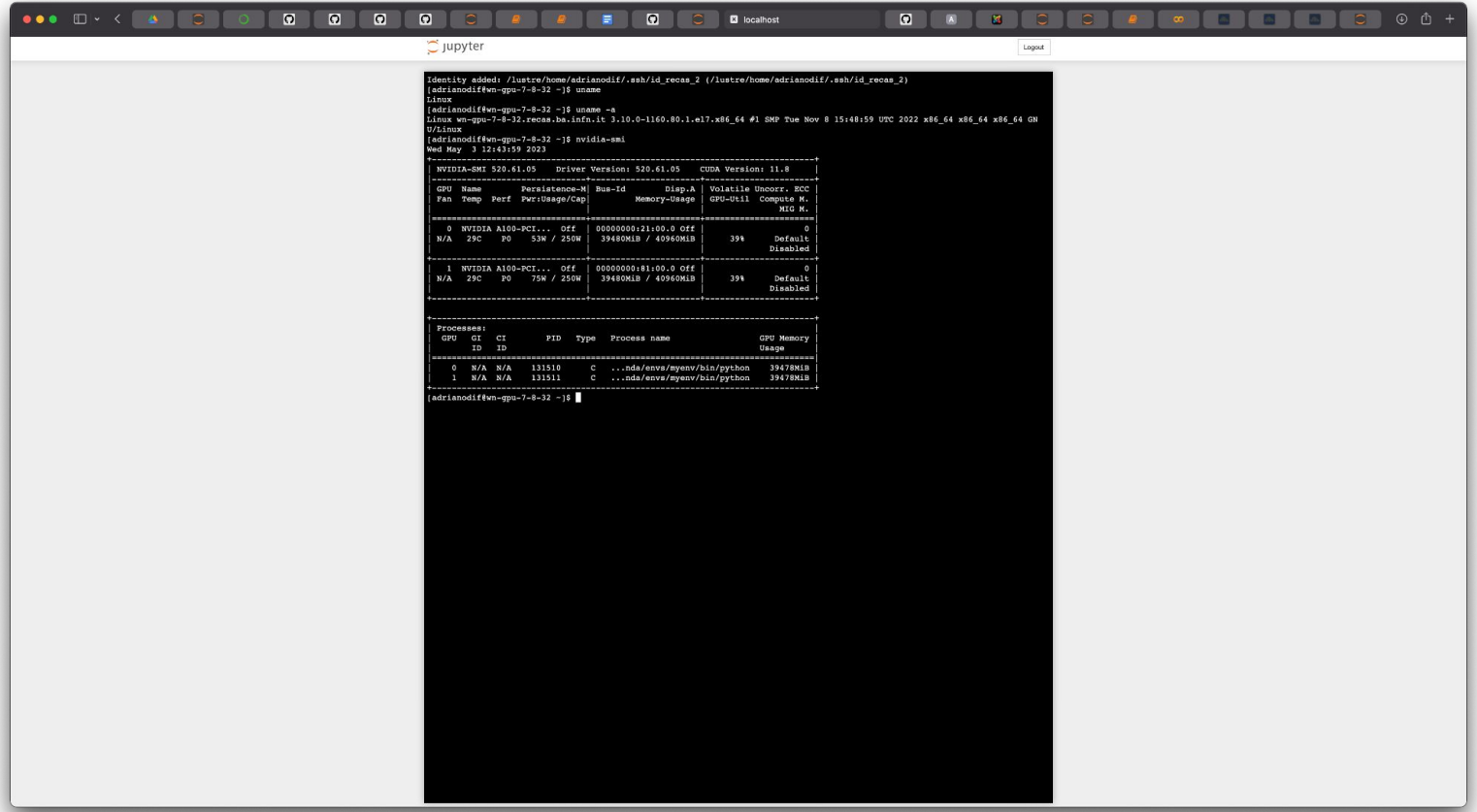
Running on ReCaS

- Let's open a **terminal**.



Running on ReCaS

- Running a **terminal** from a **browser** on the remote machine.



Decision Trees

Adriano Di Florio (INFN & Politecnico Bari)

4 May 2023

Terminology

Test data

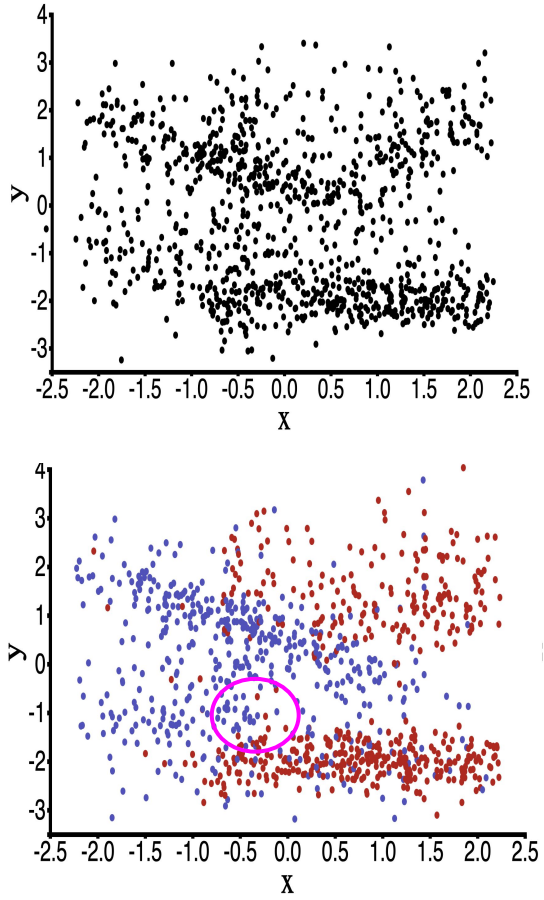
The ultimate goal of building a classifier is to be able to use it on previously unseen data and recover the correct classifications for each data sample. This unseen dataset is typically referred to as your *test data*.

Training data

In order to build/train your classifier, you will need to provide a library of examples of each target class. For supervised learning, this dataset must be pre-labelled and it is typically referred to as your *training data*.

Validation data

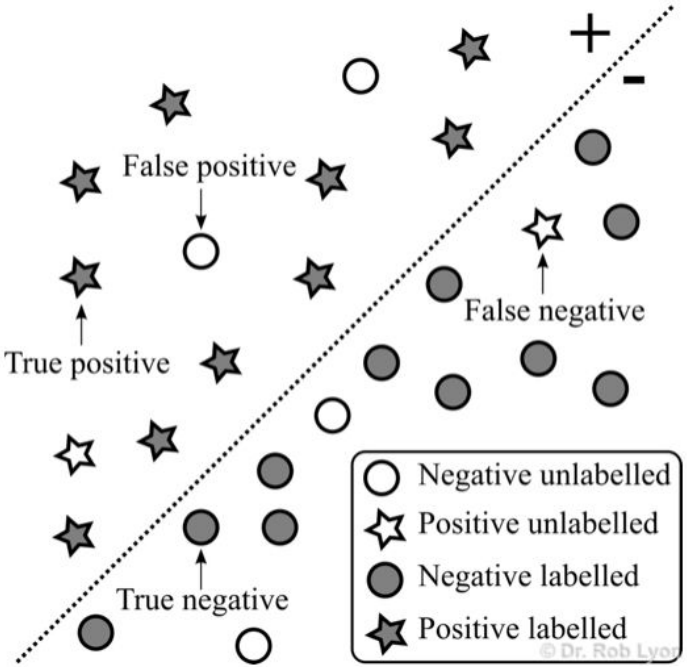
The validation dataset is a **subset of your training data**. It is not your test data. You must not use your test data to train your classifier in any way.



Metrics

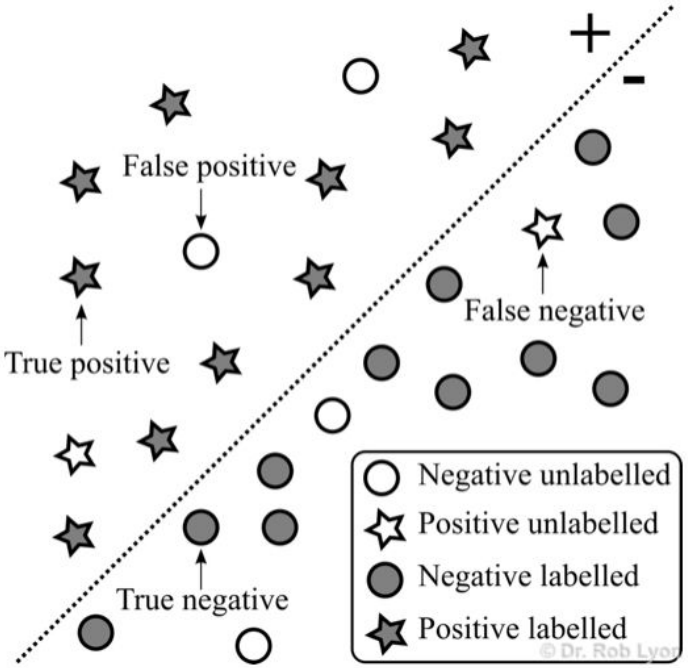
- There are a variety of ways to evaluate the performance of a machine learning model. Which one you choose should depend on the objective of your classification. Before we look at some common performance metrics we first need to define a few terms.

- Suppose we have two target classes; these could be *cat* and *mouse*, or alternatively *pulsar* and *non-pulsar*, but here I'm just going to call them *positive* and *negative*. When we apply a machine learning model to the unlabelled test data composed of these classes it fits a split that looks like this:



Metrics

- An unlabelled (test) data sample from class one that has been correctly labelled is called a ***true positive***, but a sample that has been incorrectly labelled is called a ***false negative***; likewise, an unlabelled (test) data sample from class two that has been correctly labelled is called ***true negative***, and a sample that has been incorrectly labelled is called a ***false positive***.



- Some commonly used performance metrics are:

$$precision = \frac{T_P}{T_P + F_P}$$

$$F1 = 2 \times \frac{precision \times recall}{precision + recall}$$

$$recall = \frac{T_P}{T_P + F_N}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Confusion Matrix

- An unlabelled (test) data sample from class one that has been correctly labelled is called a ***true positive***, but a sample that has been incorrectly labelled is called a ***false negative***; likewise, an unlabelled (test) data sample from class two that has been correctly labelled is called ***true negative***, and a sample that has been incorrectly labelled is called a ***false positive***.

- We use these names to describe the different types of errors and hence the performance metrics of the machine learning model.

	Truth	
Prediction	Positive	Negative
Positive	True Positive (TP)	False Positive (FP) Type I Error
Negative	False Negative (FN) Type II Error	True Negative (TN)