

# Installation instructions

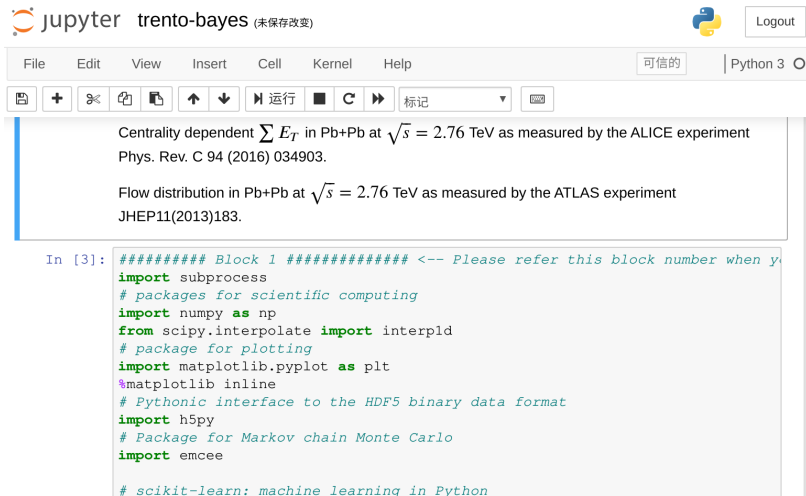
Instructions: <https://github.com/kewei Yao/JETSCAPE2020-TRENTO-BAYES>  
These examples do not require docker. They are contained in Jupyter notebooks.

Once you follow through the instructions. Activate the conda virtual environment and open the Jupyter notebook. There are three separate Jupyter notebooks in the folder. Also make sure you have the data folder “ModelData”.

	Name	Last Modified	File size
<input type="checkbox"/>	0		
<input type="checkbox"/>	/		
<input type="checkbox"/>	ExpData	10 天前	
<input type="checkbox"/>	ModelData	6 天前	
<input type="checkbox"/>	SimpleGaussianProcess.ipynb	5 小时前	78.7 kB
<input type="checkbox"/>	SimplePCA.ipynb	4 小时前	280 kB
<input type="checkbox"/>	trento-bayes.ipynb	5 小时前	400 kB
<input type="checkbox"/>	data.tar.gz	5 天前	209 MB
<input type="checkbox"/>	environment.yml	7 小时前	216 B
<input type="checkbox"/>	postBuild	6 小时前	118 B
<input type="checkbox"/>	README.md	5 小时前	4.28 kB
<input type="checkbox"/>	requirements.txt	7 小时前	14 B

## Load and test the libraries

Open the notebook “trento-bayes.ipynb”. Run the code blocks by pressing “Shift+Enter”. Check if you can run the following block that loads all the modules to be used in the exercises.



The screenshot shows a Jupyter Notebook window titled "trento-bayes (未保存改变)". The interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Help), a toolbar with icons for file operations and execution, and a code editor. The code editor contains the following Python code:

```
In [3]: ##### Block 1 ##### <-- Please refer this block number when y
import subprocess
# packages for scientific computing
import numpy as np
from scipy.interpolate import interp1d
# package for plotting
import matplotlib.pyplot as plt
%matplotlib inline
# Pythonic interface to the HDF5 binary data format
import h5py
# Package for Markov chain Monte Carlo
import emcee

# scikit-learn: machine learning in Python
```