# PyMC3 Set-up
# QED Supergroup Workshop 1/31/22
# by Jacob Stanley

**Some troubleshooting comments:**
The below steps should work on a Linux system (and I assume a Mac). However, if your system does not have `conda` installed or a suitable C-compiler you will need to install those prior to continuing with the following steps…

**The following steps will help you to set up a virtual environment, install PyMC3 and configure that environment to be run within a Jupyter notebook.**

**Create a new conda environment:**
In order to run PyMC3 we will first create a new conda environment in which all necessary packages will be installed. We will run everything in Python 3.9. Run the following at the command line, which will create the environment with the name "pymc."

```
(base) $ conda create -n pymc python=3.9

…

Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#     $ conda activate pymc
#
# To deactivate an active environment, use
#
#     $ conda deactivate

(base) $
```

**Activate the conda environment:**

```
(base) $ conda activate pymc
(pymc) $
```

**Install supporting packages first:**
The math kernel library is a required precursor for PyMC3 and we will use Jupyter notebooks to demo the package.

```
(pymc) $ conda install mkl-service jupyter
…
(pymc) $
```

**Install PyMC3 into the environment:**
Install it from the conda-forge channel.

```
(pymc) $ conda install -c conda-forge pymc3
…
(pymc) $
```

**Confirm that PyMC3 was installed successfully:**
In order to confirm correct installation, you should run a basic model within the Python interpreter. So so running the following lines: start the python interpreter, import the pymc3 library, and create the below model within the “with” code block. If installed correctly, you should see the subsequent lines at your command prompt.

```
(pymc) $ python
Python 3.9.7 (default, Sep 16 2021, 13:09:58)
[GCC 7.5.0] :: Anaconda, Inc. on linux
Type "help", "copyright", "credits" or "license" for more information.

>>> import pymc3 as pm
>>> with pm.Model() as mod:
...     a = pm.Uniform('a', lower=10, upper=20)
...     samp = pm.sample(100, step=pm.Metropolis())

Only 100 samples in chain.
Multiprocess sampling (4 chains in 4 jobs)
Metropolis: [a]
Sampling 4 chains for 1_000 tune and 100 draw iterations (4_000 + 400 draws
total) took 1 seconds.% [4400/4400 00:00<00:00 Sampling 4 chains, 0
divergences]
The rhat statistic is larger than 1.05 for some parameters. This indicates
slight problems during sampling.
>>>
```

**Install ipykernel:**
The ipykernel will allow you to access the pymc conda environment from the jupyter notebook.
Install it from the anaconda channel.

```
(pymc) $ conda install -c anaconda ipykernel
…
(pymc) $
```

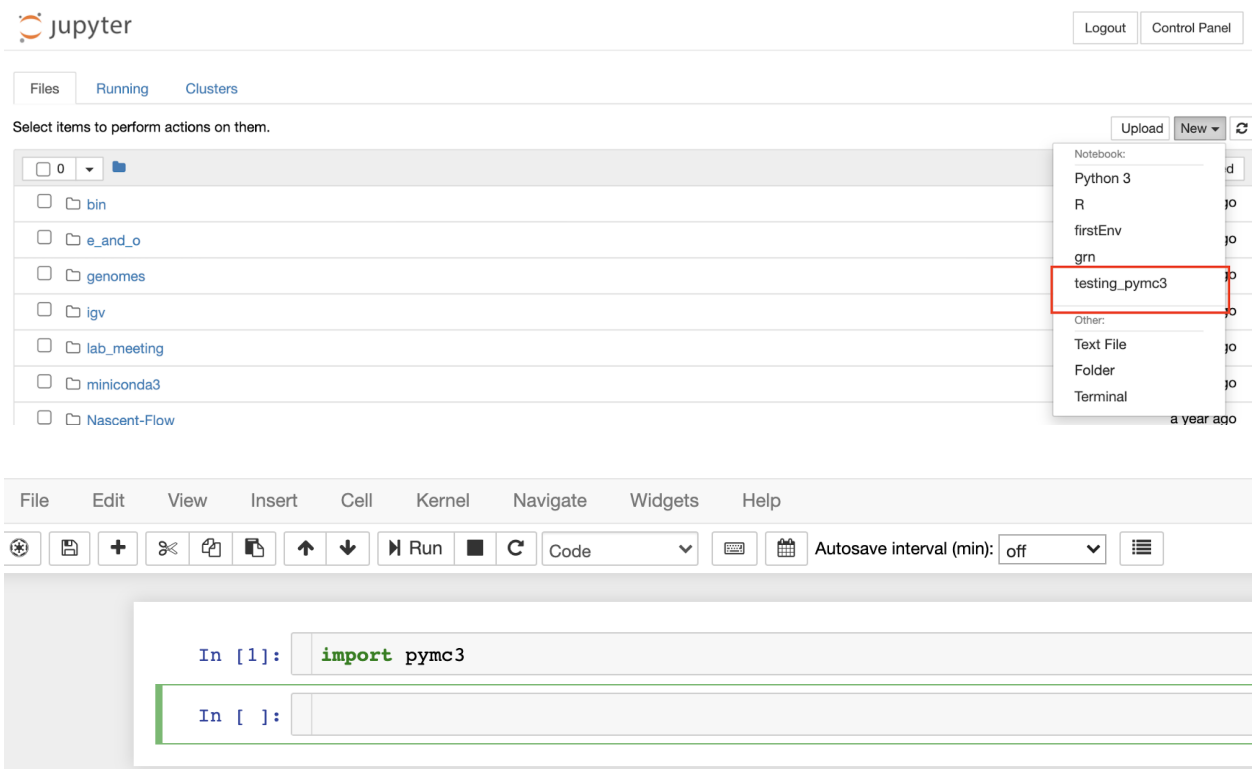
**Activate the jupyter kernel:**

```
(pymc) $ python -m ipykernel install --user --name=pymc
Installed kernelspec pymc in /home/user/.local/share/jupyter/kernels/pymc
(pymc) $
```

**Open a jupyter notebook:**

Within the jupyter server you should be able to open a new notebook with the specified environment (environment named “test_pymc3” instead of “pymc” in the below image). With the notebook open confirm you can import pymc3 (second image).

```
(pymc) $ jupyter notebook
```




You’ve installed PyMC3 to be run within the Jupyter notebook. This will be the starting point for the workshop.