Introduction to Numpy & Scipy

Wim R.M. Cardoen, PhD
University of Utah
Center for High Performance Computing
155 S. 1452 E., Rm 420
Salt Lake City
UT, 84112
USA
`wim.cardoen@utah.edu`

May 19, 2020

Introduction

In this document we describe the installation of the tutorial/lecture notes and/or verification of the required software.

I Installation of the tutorial

The tutorial/lecture notes can be retrieved as follows:

- either by using Git[1]:
  git clone https://github.com/wcardoen/numpy-rmacc2020.git
- by downloading the following zip file:
  https://github.com/wcardoen/numpy-rmacc2020/archive/master.zip

In the former case the local directory `numpy-rmacc2020` will be created.

In what follows we assume that the newly created directory bears the name `numpy-rmacc2020`.

[1] In this case we presume that the Git software is installed on your machine. The Git software can be freely obtained from the following site: https://git-scm.com/

II Required software

- Python3
  We will be using Python3 during the workshop. We also require the following Python packages to be installed:
  - ⋆ numpy >= 1.12.0
  - ⋆ scipy >= 0.19.0
  - ⋆ matplotlib >= 1.5.0
  - ⋆ notebook >= 4.2.0
- Jupyter
  The lectures will be presented using Jupyter notebooks.

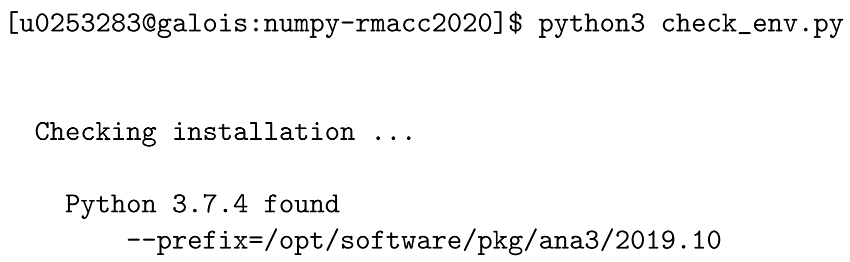
In the tutorial materials there is a script `check_env.py` (see Section III) that can be used to check your installation.

III Check your Python installation

The packages that are required for this tutorial are frequently available on most OSs. Therefore, it is quite likely that you don't have to install any additional packages.

After installing the tutorial, you can immediately check whether the required packages are present on your system.

```
$ cd numpy-rmacc2020[2]
$ python3 check_env.py[3]
```



```
[u0253283@galois:numpy-rmacc2020]$ python3 check_env.py


  Checking installation ...

    Python 3.7.4 found
        --prefix=/opt/software/pkg/ana3/2019.10
```

[2] The lines starting with a $ sign are commands to be executed in a Shell.

[3] The output generated by this command will be dependent on the installed version of Python and the required packages.

```
            numpy  inst.:    1.17.2  >= req.:    1.12.0 -> OK
            scipy  inst.:     1.3.1  >= req.:    0.19.0 -> OK
       matplotlib  inst.:     3.1.1  >= req.:     1.5.0 -> OK
         notebook  inst.:     6.0.1  >= req.:     4.2.0 -> OK

    All required packages are installed
    Congratulations!

  End checking installation
```

You can verify the presence of Jupyter like this:

```
$ jupyter --version[4]
```

```
[u0253283@galois:numpy-rmacc2020]$ jupyter --version
jupyter core     : 4.5.0
jupyter-notebook : 6.0.1
qtconsole        : 4.5.5
ipython          : 7.8.0
ipykernel        : 5.1.2
jupyter client   : 5.3.3
jupyter lab      : 1.1.4
nbconvert        : 5.6.0
ipywidgets       : 7.5.1
nbformat         : 4.4.0
traitlets        : 4.3.3
```

If some of the above packages are missing you can proceed with the installation step (section IV). Otherwise you are completely ready for the tutorial.

IV Installation of the required software

If you don't have Python3 installed or you have an incomplete installation (see section III), installing the Anaconda distribution is the easiest option.

The Anaconda distribution is available for the Mac, Windows & Linux OSs. It also contains all the aforementioned requirements/packages.

Return to section III to verify that your installation is properly working.

[4] The output generated will depend on the Jupyter version installed on your machine.