TUTORIAL

In order to reproduce or write scripts and notebooks for ATLAS data analysis using ROOT and Python, you'll need to keep on the following instructions.

Requirements

1. ROOT and TMVA.
2. Anaconda and Scikit-Learn, NumPy, Keras, Pandas, PyROOT, root_numpy, rootpy y matplotlib libraries, and metakernel.

Installation:

1. **ROOT**:
First of all you will need to instal some dependencies

```
sudo apt-get install git dpkg-dev cmake g++ gcc binutils libx11-dev
                libxpm-dev libxft-dev libxext-dev
```

Another optional dependencies

```
        sudo apt-get install gfortran libssl-dev libpcre3-dev
    xlibmesa-glu-dev libglew1.5-dev libftgl-dev libmysqlclient-dev
              libfftw3-dev libcfitsio-dev graphviz-dev
  libavahi-compat-libdnssd-dev libldap2-dev python-dev libxml2-dev
                libkrb5-dev libgsl0-dev libqt4-dev
```

In order to use ROOT and Python 3, it's required to have Python 3 as your default Python version.

Create a new folder called *root*, and inside the new folder, clone the repository of ROOT with git

```
git clone http://github.com/root-project/root.git
```

Checkout to the branch of the desired ROOT version

```
git checkout -b <version> <version>
```

For example: `git checkout -b v6-18-04 v6-18-04`

Create a new folder inside root, called *root-v6-18-04*, depending on the version you downloaded. Go to *root-v6-18-04* and run

```
cmake ../root -DPYTHON_EXECUTABLE=$(which python)
```

to build ROOT. Then, go to the build folder (*root-v6-18-04*), and run

```
sudo cmake --build . --target install
```

to install ROOT. This will take a couple of hours. As last step, create the environment variables

```
export ROOTSYS=/home/username/root/root-v6-18-04
export PATH=$ROOTSYS/bin:$PATH
export LD_LIBRARY_PATH=$ROOTSYS/lib:$LD_LIBRARY_PATH
cd /home/username/root/root-v6-18-04
. bin/thisroot.sh
```

On the terminal add the source

```
source /home/username/root/root-v6-18-04/bin/thisroot.sh
```

Now you can run ROOT by taping `root` on the terminal.

TMVA is included by default in ROOT installation. You can find ROOT build requirements in: https://root.cern.ch/build-prerequisites#ubuntu

2. **Anaconda**

Download the latest version of Anaconda from https://www.anaconda.com/download/, unzip the compressed file and run the bash file:

```
bash AnacondaX-Y.Z.W-Linux-x86_64.sh
```

Once the installation has finished, you have to activate the installation:

```
source ~/.bashrc
```

All done! You can test the installation with:

```
conda list
```

Anaconda includes numpy, pandas, and matplotlib by default, but you'll have to install **scikit-learn**, **keras**, **pandas**, **rootpy**, **root_numpy** and **metakernel** packages. This can be done by running

```
pip install -U <package_name>
```

or

```
conda install <package_name>
```

For using `pip` (for python) or `pip3` (for python 3), you'll need to install it

```
apt get install pip
apt get install pip3
```

Execution:

You can run jupyter notebook with ROOT kernel by executing: `root --notebook` in your terminal.

Connection to computational clusters

1) Write on your terminal for connecting to **lxplus** (CERN cluster):

```
ssh <user_name>@lxplus.cern.ch
```

and for connecting to **hpc** (UTFSM cluster):

```
ssh <user_name>@ui.hpc.utfsm.cl
```

Enter your password and you will get connected.

2) Then, open a new file, name it *.bashrc* or *bash_login* file and type the next lines

```
#export ROOTENV_NO_HOME=1
export
ATLAS_LOCAL_ROOT_BASE=/cvmfs/atlas.cern.ch/repo/ATLASLocalRootBase
alias setupATLAS='source
${ATLAS_LOCAL_ROOT_BASE}/user/atlasLocalSetup.sh'
```

3) In your home, create a directory called *WorkArea,* and inside create two more directories: *run* and *build*. Then, copy the test.tar.gz file from Edson's local and paste it on your WorkArea.

```
cp /user/e/edson/dihiggs/test.tar.gz your/WorkArea/
```

Untar the file

```
tar -zxvf test.tar.gz
```

Rename the folder of the test files

```
mv test source
```

4) Go to the *build* directory and run the following

```
cd ../build
setupATLAS
lsetup asetup
release=`cat ../source/CxAODBootstrap_VHbb/bootstrap/release.txt`
echo "release=$release"
asetup AnalysisBase,$release,here
cp CMakeLists.txt ../source
cmake ../source
cmake --build .
source x86_64-slc6-gcc62-opt/setup.sh
```

5) Execute from the run directory

```
hsg5framework
```

It will crash if you don't have data, so the next step will be to copy some simulated data from lxplus Cluster.

Useful tutorials

1. **ROOT:** http://physics.bu.edu/neppsr/2007/TALKS-2007/ROOT_Tutorial_Bose.pdf.
2. **BDT score**: http://software.icecube.wisc.edu/documentation/projects/pybdt/man_bdt_intro.html
3. **Histogram Analysis:** https://cheatham1.gitbooks.io/get-started/content/the_display_histograms.html
4. **TMVS User guide:** **https://root.cern.ch/root/html/guides/tmva/TMVAUsersGuide.pdf**
5. **Multivariate Analysis Tutorial (complete sequence using TMVA)** https://agenda.infn.it/event/13733/contributions/20520/attachments/14642/16541/MVATutorial.pdf
6. **Keras Tutorial (using TMVA too)** https://github.com/stwunsch/iml_keras_workshop/blob/master/slides/slides_iml_keras_workshop.pdf
7. **scikit-learn for TMVA Users** https://betatim.github.io/posts/sklearn-for-TMVA-users/