

Pocket-Coffea Brief Read Me -Polytimi

Configuration File Setup

To get started, create a configuration (cfg) file with the following components :

1. Datasets
2. Workflow
3. Skimming
4. Preselection
5. Categories for data division
6. Weights or scale factors (sfs) to apply
7. Variations
8. Histograms to plot
9. Executor configuration (e.g., dask at lxplus)

Example cfg files can be found at the following links:

- [My files](#) on Github
- [Example](#) from the Pocket Coffea team

Or at the pwd:

- /afs/cern.ch/user/p/piosifid/public/forNiki/AnalysisConfigs/configs/ttHbb/dilepton

[Detailed instruction can be found here](#)

Initial Setup

1. *Initialize VOMS Proxy:*

```
voms-proxy-init -voms cms
```

2. *Enter the Apptainer Image:*

```
apptainer shell -B /afs -B /cvmfs/cms.cern.ch \ -B /tmp -B /eos/cms/ -B /etc/sysconfig/ngbauth-submit \ -B ${XDG_RUNTIME_DIR} --env KRB5CCNAME="FILE:${XDG_RUNTIME_DIR}/krb5cc" /cvmfs/unpacked.cern.ch/gitlab-registry.cern.ch/cms-analysis/general/pocketcoffea:lxplus-el9-latest (or -stable instead of latest)
```

Datasets

To create datasets, build a datasets.json file ([example](#)), then run the following command:

- `pocket-coffea build-datasets --cfg myDE.json -bs T1_DE_KIT_Disk -bs T1_FR_CCIN2P3_Disk -bs T1_IT_CNAF_Disk -bs T1_IT_CNAF_Tape -bs T1_RU_JINR_Disk -bs T1_UK_RAL_Disk -bs T1_US_FNAL_Disk -bs T2_RU_INR -bs T2_UA_KIPT -bs T2_CH_CSCS -bs T2_US_MIT -bs T2_BR_SPRACE -bs T2_US_Vanderbilt -bs T2_DE_DESY -bs T2_IT_Bari -bs T2_IT_Legnano -bs T2_IT_Pisa -bs T2_IT_Rome`

This command assumes that the datasets.json is the myDE.json from my github (for the egamma datasets) also includes examples of T2 sites that have been blacklisted. (you can also use `-ws T2_IT_Legna` to whitelist a place)

There are already available datasets [here](#)

Running the Configuration

To run your cfg file, execute:

- `pocket-coffea run --cfg cfg_2023_postBPix_DE.py -o output_2023_postBPix_DoubleEle -ro params/executor_options.yaml -e dask@lxplus -s 150`
- `pocket-coffea run --cfg MyCFG.py -o output_v1 -e dask@lxplus`

Important! Before running on dask at lxplus one can check the cfg for errors with the --test option:

- `pocket-coffea run --cfg cfg_2023_postBPix_DE.py -o output_177_4 --test`

For more available functions, use:

- `pocket-coffea --help`

For example, to see available keys and functionalities for running, use:

- `pocket-coffea run --help`

Plotting Results

To plot the results go to your output directory (e.g. output_v1), use:

- `pocket-coffea make-plots -i output_all.coffea --cfg parameters_dump.yaml -v 15 --overwrite`
-

Usefull links:

Pocket_Coffea_ReadtheDocs:

<https://pocketcoffea.readthedocs.io/en/stable/index.html>

Killians and Phillip directory:

https://gitlab.cern.ch/phnattla/ttHbbPocketConfigs/-/tree/Kilian/Run3/DL?ref_type=heads

My directory on github: (Analysis config, Dilepton branch)

<https://github.com/piosifid/AnalysisConfigs/tree/Dilepton/configs/ttHbb/dilepton>

Analysis Config directory (where different analyses can be found ...we belong to the config/ttHbb/dilepton)

<https://github.com/mmarchegiani/AnalysisConfigs>

Pocket Coffea -latest codes

<https://github.com/PocketCoffea/PocketCoffea/tree/main>

For a quick first run do:

```
cd /afs/cern.ch/user/p/piosifid/public/forNiki/AnalysisConfigs/configs/ttHbb/dilepton
```

```
voms-proxy-init -voms cms
```

```
apptainer shell -B /afs -B /cvmfs/cms.cern.ch \ -B /tmp -B /eos/cms/ -B  
/etc/sysconfig/ngbauth-submit \ -B ${XDG_RUNTIME_DIR} --env  
KRB5CCNAME="FILE:${XDG_RUNTIME_DIR}/krb5cc"  
/cvmfs/unpacked.cern.ch/gitlab-registry.cern.ch/cms-analysis/general/pocketcoffea:lxplus-  
el9-latest
```

```
pocket-coffea run --cfg MyCFG.py -o output_v1 --test
```

```
pocket-coffea run --cfg MyCFG.py -o output_v1 -e dask@lxplus
```

```
cd output_v1
```

```
pocket-coffea make-plots -i output_all.coffea --cfg parameters_dump.yaml -v 15 --overwrite
```

