1. Git clone https://github.com/CaloChallenge/homepage.git
2. In the homepage folder there is another folder called code where you will find evaluate.py. We have to run this file.
3. To run this file I am using the previous conda environment I created for CaloFlow.
4. This code also requires the installation of latex for plotting.
5. To do the installation of latex, you have to follow the instructions mentioned below:
    a. This code uses Latex with Matplotlib. Since I need to install TeXLive. TeXLive is available in the modules of Rivanna. Here is the link to do that https://www.rc.virginia.edu/userinfo/rivanna/software/texlive/#add-local-package
    b. In this link it mentioned to create a folder like this ~/texmf/tex/latex and put your required package from the https://ctan.org/
    c. I found that I needed type1ec and type1cm , I copied the sty file of type1ec from https://ctan.org/
    d. I also copied the type1cm and base folder found in `$EBROOTTEXLIVE/texmf-dist/tex/latex` Into ~/texmf/tex/latex this directory.
    e. After following these instructions in https://www.rc.virginia.edu/userinfo/rivanna/software/texlive/#add-local-package

    I got another error message saying `FileNotFoundError: [Errno 2] No such file or directory: 'dvipng'`

    Since I also followed the instructions of this link https://hub.docker.com/r/uvarc/dvipng

    f. Here in the link they mentioned “module load singularity” instead of singularity I used ‘module load apptainer’
6. For this I wrote the slurm job script this way:

```
#!/bin/bash

#SBATCH -N 1

#SBATCH -n 1

#SBATCH --job-name=caloChallenge

#SBATCH -t 50:00:00

#SBATCH --mem=64000

#SBATCH -p bii-gpu

#SBATCH --gres=gpu

#SBATCH -A bii_nssac
```

```
module load anaconda/2023.07-py3.11
conda activate caloflow

module load texlive/2023
export PATH=~/bin:$PATH
module load cuda/12.2.2
module load cudnn/8.9.4.25
module load apptainer

#python updated_evaluation.py
#python src/main.py params/pions.yaml -c

python evaluate.py -i
'/scratch/fa7sa/IJCAI_experiment/Generated_shower/CaloDiffusion_10000_samp
le/test_ds2.h5' -r '/scratch/fa7sa/IJCAI_experiment/dataset_2/dataset_2_2.hdf5'
-m 'hist-p' -d '2' --output_dir 'evaluation_results/'

I will be attaching the script with the email.
```

The following information is required for arguments.

```
""" Main script to evaluate contributions to the Fast Calorimeter Challenge 2022

    input:
        - set of events in .hdf5 file format (same shape as training data)
    output:
        - metrics for evaluation (plots, classifier scores, etc.)

    usage:
        -i --input_file: Name and path of the input file to be evaluated.
```

-r --reference_file: Name and path of the reference .hdf5 file. A .pkl file will be
created at the same location for faster subsequent evaluations.

-m --mode: Which metric to look at. Choices are

'all': does all of the below (with low-level classifier).

'avg': plots the average shower of the whole dataset.

'avg-E': plots the average showers at different energy (ranges).

'hist-p': plots histograms of high-level features.

'hist-chi': computes the chi2 difference of the histograms.

'hist': plots histograms and computes chi2.

'cls-low': trains a classifier on low-level features (voxels).

'cls-low-normed': trains a classifier on normalized voxels.

'cls-high': trains a classifier on high-level features (same as histograms).

-d --dataset: Which dataset the evaluation is for. Choices are

'1-photons', '1-pions', '2', '3'

--output_dir: Folder in which the evaluation results (plots, scores) are saved.

--save_mem: If included, data is moved to the GPU batch by batch instead of once.
This reduced the memory footprint a lot, especially for datasets 2 and 3.

--no_cuda: if added, code will not run on GPU, even if available.

--which_cuda: Which GPU to use if multiple are available.

additional options for the classifier start with --cls_ and can be found below.

"""