Project 3 Group 4

CHALLENGE 1- Reproducibility Challenge

Shreya Verma:

1. STEP 1

Forking the repository from GitHub ClimSim Page.

2. STEP 2

Downloaded the sub-sampled low resolution real geography version of the data from HuggingFace in my local environment.

Estimated Time Taken: 20 minutes.

3. STEP 3

Worked on Jupyter Notebook for reproduction of quickstart demo notebook. Install the climsim_utils python tools by running the following code `pip install .` for the first time, and when all necessary dependencies were installed, used `pip install . --no-deps` for all following runs.

Estimated Time Taken: 5 minutes.

4. STEP 4

- Importing the training and validation data. Changed file paths according to the local environment.
  Error: The kernel appears to have died. It will restart automatically.
  Inference: Discovered that data files got downloaded but couldn't load because of file size and the system kept crashing.
  Steps Taken: A team member managed to take a subsample of each of the downloaded files.

Estimated Time Taken: 5 minutes.

5. STEP 5

- The small size files were imported. Loading them created the following issue.
  Error: All functions from data_utils applied to data_split = 'val' were giving dimension errors.
  Inference: Discovered that issue was related to number of rows in dataframe not being a multiple of 384 in smaller .npys
  Steps Taken: A team member made the subsample the files of size that's a multiple of 348.

6. STEP 6

Training, Validation, and Evaluation worked without any Errors.

Estimated Time Taken: 15 minutes.

Puqi Song:

7. STEP 1

    Local computing environment: RAM:13.7 GB

    Google Colab: RAM: 12.7 GB

8. STEP 2

    Forking the repository from GitHub ClimSim Page.

9. STEP 3

    Downloaded the six datasets in .npy format.

    Estimated time:15 min

    Uploaded the six datasets into Google Drive.

    Estimated time:3 hours

10. STEP 4

    Tried to load the datasets in Google Colab, but failed.

    Reason: The datasets were too big to be loaded.

    Solution: Tried to load the subset of the datasets that were extracted by our group member Michael.

    Result: Succeeded.

11. STEP 5

    Reproduced the whole quickstart notebook in Google Colab.

    Time taken: 33 seconds(from pip install . to the last step).

12. STEP 6

    The code runs without error, but the figures of (c) R2 are blanket.

Michael Wiley :

13. STEP 1

Download prerequisites

Estimated Time: About 2 hours on my end, but your mileage will vary depending on your internet speed. Mine is very slow.

- Forked ClimSim repository https://github.com/leap-stc/ClimSim

  (Cloning is okay too, I forked in case I end up wanting to submit a PR to the main repo sometime in the future.)

- Downloaded required data in preferred format.
  https://huggingface.co/datasets/LEAP/subsampled_low_res/tree/main

  Either all of the .npy files or all of the .parquet files - Functionally and size-wise there's little difference between them. I recommend .npy as that's what the demo notebook expects as is.

14. STEP 2

File organization

Estimated Time: 5 minutes

I organized my project in the following way:

ClimSim/

├── demo_notebooks/

├──────.ipynb files

data/

├── .npy files

Separating the repository from the data was a personal choice and anyone reproducing these steps is free to place it wherever they choose, just be aware that it will affect file paths in the notebook (see step 6).

15. STEP 3

Check System Requirements

Estimated Time: 5 minutes

By default, the demo notebook requires loading around 12 GB of data into memory (but uses more at points due to an expensive matrix inversion)

I was *barely* able to reproduce the notebook in its entirety with a desktop with the below specs. It completely froze up for minutes at a time at some points. During the most intense sections, my monitor would not update and I could not move my mouse until the cell completed. I was also monitoring my resource usage and sometimes my RAM would cap out and flow into my CPU - causing that to max out as well.

CPU: Intel64 Family 6 Model 63 Stepping 2, GenuineIntel

CPU Cores: 6

CPU Frequency: 3301.0 MHz

RAM: 15.927257537841797 GB

Virtual Memory: 42.31984329223633 GB

GPU: NVIDIA GeForce GTX 970, Memory: 4.0 GB

I created a very simple Python script to extract the above information if others would like to compare their specs to mine:

https://github.com/LumineonRL/GetSysInfo/blob/main/GetSysInfo.py

I'd love to know if anyone successfully ran the full notebook with specs even slightly worse than mine.

16. STEP 4

Install dependencies

Estimated Time: 5 minutes

Opened a terminal and `cd`'d into the root of `ClimSim` and ran `pip install .` to install the package dependencies.

Note that I use VSCode as my Python/Jupyter editor where opening a terminal to run that command is straightforward. I'm actually not sure how you'd do this if you were using the traditional browser version of Jupyter without interacting with a terminal.

17. STEP 5

Open the notebook

Estimated Time: 1 minute

The file we're tasked with running is located in `ClimSim/demo_notebooks/quickstart_example.ipynb`

18. STEP 6

Update hard-coded paths.

Estimated time: 10 minutes if done from scratch, 2 minutes if following my solution

I'm not a fan of hard-coding file paths into code and the default notebook has a few of these.

Each of these variables is suffixed with `_path` if you want to find them and replace them yourself.

If you used the same directory structure as me in step 2, you can simply copy my modified code that uses relative file paths.

https://github.com/LumineonRL/ClimSim/blob/main/demo_notebooks/quickstart_example.ipynb

Note that both the original repository's notebook and my version assume that the .npy version of the files was downloaded. If you downloaded .parquet instead you will need to manually update that.

19. STEP 7

Manual Garbage Collection

Estimated time: N/A

If your machine is considerably more powerful than mine feel free to skip this step entirely.

Although Python does a fantastic job of garbage collection on its own, running this notebook was a rare instance where every single bit of memory mattered. As such, I inserted `gc.collect` calls before and after some of the more expensive calls to free up as much space as possible.

If you feel it may help, you can find my garbage collection locations in my version of the startup notebook https://github.com/LumineonRL/ClimSim/blob/main/demo_notebooks/quickstart_example.ipynb

20. STEP 8

Run the Notebook

Estimated time: 15 minutes

Running the entire notebook from start to finish took approximately 15 minutes. A detailed breakdown of how long each component took can be found below. A more powerful machine should be able to run through this considerably faster.

Cell	Time (Seconds)
Import data_utils	9.4
Instantiate class	0.1
Load training and validation data	67.6
Train models	0.8
Add bias unit	174.5
GC	2.2
Create model	461
Set pressure grid	4.2
Load predictions	1.9
Weight predictions	16.7
Set and calculate metrics	14
Create plots	1
Load scoring data	4.4

Set pressure grid	1.7
Load predictions	30.3
Weight predictions and target	82.8
GC	0
Create plots	1
Sum	873.6

21. STEP 9

Create Smaller Dataset

Estimated time: N/A

This isn't required, but my version of the notebook from Step 6 also contains a cell to save smaller versions of the dataset that other team members with weaker machines can run.

The important bit is that the smaller dataset *must* have a number of rows that is a factor of 384 in order for the `set_pressure_grid` cells to run. I intend to look more into this error to see if anything can be done to either make that function more robust, or make it clearer to the user that data that is not a correct multiple will cause issues.

Finally, a notebook intended to be ran by users with weaker machines using the smaller datasets produced above was created.

This notebook can be found
https://github.com/LumineonRL/ClimSim/blob/main/demo_notebooks/quickstart_example_small_data.ipynb

Please attempt to run this one in the event that your machine can not run the original.

Zhenhui Wang:

22.STEP 1

Download the subsampled low-resolution real-geography version of the data.

Estimated time:2hrs

23.STEP 2

Fork the repo from GitHub.

Estimated time:15 min

24.STEP 3

Install the climsim_utils python tools by running the following code pip install.

Problem: ERROR: Could not find a version that satisfies the requirement climsim_utils (from versions: none) ERROR: No matching distribution found for climsim_utils Note: you may need to restart the kernel to use updated packages.

Solution: I used the colab instead. After I changed my path to the path of climsim, the pip install . worked.

Estimated time: 6hrs

25.STEP 4

- Imported the training and validation data.
  Error: The kernel appears to have died. It will restart automatically.
  Solution: A team member managed to take a subsample of each of the downloaded files with smaller size. And the smaller size files could be loaded.

Estimated time:20 min

26.STEP 5

- The small size files were imported and loaded.

Estimated time:5 min

27.STEP 6

Training, Validation, and Evaluation worked without any Errors.

Estimated time:5 min

Daniel Lam:

28.STEP 1

Download subsampled low-resolution data processed by Michael

Estimated time: 10mins

29.STEP 2

Fork the repo from GitHub and organize using anaconda and jupyter notebook

Estimated time:15 min

30.STEP 3

Install climsim_utils tools by using pip in the root of my forked github repo. Did this through the terminal.

Estimated time:10 min

31.STEP 4

Instantiate data class using code from the quickstart_example notebook

Estimated time:5 min

32.STEP 5

Adjust file paths and load data using the data class and data loading functions.

Estimated time:5 min

33.STEP 6

Training and validation of models - went without errors

Estimated time:10 min

34.STEP 7

Model evaluation and creating plots using the code from the quickstart_example notebook, no errors

Estimated time:10 min