STAT GR5243 Project 3 Team 5 - Challenge I Reproduction Report

- **Local Computing Environment:**

	Yufei	**Shaohuan**	**Yoojin**	**Zhaolin**
Operating System	Windows 11	macOS 13.5	macOS 14.0	Windows 10
Python Version	3.10.12	3.7.13	3.9.18	3.10.
Jupyter Notebook Version	6.5.5	Google Colab Pro	6.5.4	Visual Studio
CPU	Intel Core i7-9750H	Intel Xeon E5-2620 v4	Apple M1 Pro (10 core)	Intel (R) Core (TM) i7-9750H CPU
RAM	16 GB	51GB	32GB	16G

- **Steps Taken for Successful Reproduction:**

1. Cloning the ClimSim Repository

We cloned the ClimSim GitHub repository to the local machine by using the following command:

```
!git clone https://github.com/leap-stc/ClimSim.git
```

The estimated time is 1 minute.

2. Installation of Dependencies

We installed the climsim_utils Python packages using pip. We need to run pip installation code from the root of the ClimSim GitHub repository.

The estimated time is 1 minute.

3. Importing the packages in Jupyter Notebook

After we successfully installed the python packages, we need to import this in our Jupyter Notebook. If the package is successfully installed, it should run.

4. Data Preparation

We downloaded the subsampled low-resolution real-geography dataset from the ClimSim website, and then uploaded the dataset to the correct directory.

The estimated time is quite different for members in our group, since it depends on the internet speed and computer settings. For people using Jupyter Notebook, the estimated time has taken about 25 minutes(including 15 minutes for downloading and 10 minutes for uploading), and for some, it took about 3 minutes to upload it. For people using Colab, they needed to upload the dataset to Google Drive, which took almost 3 hours.

5. Load the Dataset

In our Jupyter Notebook or google colab, we need to set the relevant file path to our corresponding directory. We had to make sure that we put the proper data path, so that we can load the dataset successfully.

6. Run the model

To make sure this Jupyter Notebook runs well in our local setting, we checked if the existing models run correctly. We also ensured that the predictions and visualizations from those models produced the output.

- **Issues Encountered:**

1. Issue with installing ClimSim packages and Environment Setting

Faced with the kernel crashing error when executing the cell. It seemed that the importing process was not successful, which might also indicate that the installing ClimSim Python packages was not successful. And not to get confused with the environment setting, we set up the virtual environment for the ClimSim project, and then the problem could be solved. Also, we need to make sure that we install the ClimSim python packages from the root of the GitHub repository.

2. Setting the proper file path

To run the guide notebook in our local settings, we had to make sure where we downloaded the dataset files in our local, and where the grid_path and norm_path were located in the cloned ClimSim Repository.

3. RAM limitation of Google Colab

When we load the dataset in Google Colab, we can not run the training model successfully because the RAM of Google Colab is not enough. So we need to upgrade to the Google Colab Pro to expand memory.

4. Upload training data to Google Drive

To run the training data in the notebook on Google Colab, we need to upload the training data to Google Drive. But it really takes time. It takes us three hours to upload the whole dataset to Google Drive so that we can run the load training and validation data section in the Google Colab.

5. Compatibility Issues

We encountered compatibility issues with TensorFlow and Python versions. After installing compatible versions of TensorFlow and Python packages, the issue was resolved.

Estimated time used for the full reproduction process:

The estimated time for the full reproduction process, including data preparation, code execution, and issue resolution, was approximately 90 minutes to 4 hours.
The time required for reproduction may vary depending on the computing environment, internet connection speed, and familiarity with the dataset and codebase. The provided estimates are approximate and may differ for other users.