extreme_rainfall_v2025

Detecting 12-hour extreme rainfall events from Stage IV data (new and improved 2025 version).

Refer to 2024 version (first release source code) or the Zenodo repository (https://doi.org/10.5281/zenodo.13286368) for the code used to generate the database analyzed in Chiappa et al. (2024).

Release Notes:
A bug was discovered in the 2024 version that caused some exceedance points to not be counted. This version uses an improved approach that is more streamlined and is far less likely to miss any points of exceedance. This is accomplished through a "merging" method instead of a replacement method when detecting simulataneous events within the specified distance threshold.

Main difference in results from 2024 version: Average annual event count over the domain went from ~67% of what is expected based on NOAA Atlas 14 thresholds to ~85%, meaning Stage IV data is better at detecting EREs than previously thought. Annual event counts are similar, but number of exceedance points and total exceedance volume is larger. Variability and trend results presented in Chiappa et al. (2024) do not change significantly.

Reference:
Chiappa, J., Parsons, D. B., Furtado, J. C., & Shapiro, A. (2024). Short-duration extreme rainfall events in the central and eastern United States during the summer: 2003–2023 trends and variability. *Geophysical Research Letters*, 51, e2024GL110424. https://doi.org/10.1029/2024GL110424

Prerequisites

- At least 1 TB of storage space recommended
- Python 3 installed with dependencies listed in code

Recommend running code using Jupyter Notebook (use .ipynb files in "notebooks" folder). Otherwise, run .py scripts in "scripts" folder.

Part I: Data Retrieval and Prep

Stage IV

1. Download Stage IV data from EOL archive https://doi.org/10.5065/D6PG1QDD
2. Extract all files that are not in .grb2 format (.Z or .gz format) and separate 01h, 06h, and 24h files into separate directories (conus only, when applicable)
3. Run `01_stageiv_renaming.py` – Renames files to keep consistent naming convention and add grb2 extension where necessary (Note: Files before 20 July 2020 are technically GRIB1 format, but adding the grb2 extension does not impact the readability of the file.)

4. Optional: Run `02_check_missing.py` and see exported csv files for times of missing Stage IV data files

NOAA Atlas 14

1. Download `NOAA_Atlas_14_CONUS.nc` from https://hdsc.nws.noaa.gov/pub/hdsc/data/tx/
2. Place in a designated folder

States

1. If not downloaded, get the ArcGIS states shapefiles (by jeff.elrod) from https://www.arcgis.com/home/item.html?id=66d95ce91afb48829a9fe9c3145c755d
2. Extract and place folder in main data directory if not already there

Tropical Cyclones

1. Download the latest IBTrACS data in csv format (`ibtracs.since1980.list.v__r__.csv`) from https://www.ncei.noaa.gov/products/international-best-track-archive "CSV Data" directory
2. Place in main data directory

Part II: Preliminary ERE Detection Algorithm

Run `03_ere_database.py` after changing paths and variables in the code (will take several hours to run – around 2 hours per year of data)

Part III: Quality Control

1. Run `04_qc_params.py` to compute relevant quality control parameters for each preliminary event. Takes several minutes.
2. Run `05_qc.py` to create and apply QC conditions. Will be applied to entire dataset regardless of resuming.

Part IV: Additional Attributes and Auto-Classification

1. Run `06_exceedance_attrs.py` to calculate attributes on the exceedance arrays. Takes several minutes.
2. Run `07_pfs.py` to obtain the 1 mm precipitation feature statistics for each hour of each event. This will take several hours to run.
3. Run `08_classification.py` to apply conditions for classifying events as tropical cyclone, isolated, MCS/synoptic, and nocturnal. Also adds attributes related to the 1 mm/hr precipitation features and diurnal cycle. (Sunset/sunrise calculations use the code from https://michelanders.blogspot.com/2010/12/calulating-sunrise-and-sunset-in-python.html saved as "sun.py" to the code directory.)
4. Run `09_final_dataset.py` to organize the data table and save as a csv to be used for analysis.

Recommend backing up all exported csv files in case of accidental replacement/altering!

Contact

Jason Chiappa - jasonchiappa2498@gmail.com

Please contact me if you have questions/issues or would like any code used to perform analysis on the database.

Project Link: https://github.com/jchiappa/extreme_rainfall

(back to top)