

Heatwave Diagnostics Package: Efficiently Compute Heatwave Metrics Across Parameter Spaces

Cameron Cummins¹ and Geeta Persad¹

¹ Department of Earth and Planetary Sciences, Jackson School of Geoscience, The University of Texas at Austin, Austin, TX, USA

DOI: [10.xxxxxx/draft](https://doi.org/10.xxxxxx/draft)

Software

- [Review](#) ↗
- [Repository](#) ↗
- [Archive](#) ↗

Editor: [Open Journals](#) ↗

Reviewers:

- [@openjournals](#)

Submitted: 01 January 1970

Published: unpublished

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).

Summary

The heatwave diagnostics package (HDP) is a Python package that provides the climate research community with tools to compute heatwave metrics for the large volumes of data produced by earth system model large ensembles, across multiple measures of heat, extreme heat thresholds, and heatwave definitions. The HDP leverages performance-oriented design using xarray, Dask, and Numba to maximize the use of available hardware resources while maintaining accessibility through an intuitive interface and well-documented user guide. This approach empowers the user to generate metrics for a wide and diverse range of heatwave types across the parameter space.

Statement of Need

Accurate quantification of the evolution of heatwave trends in climate model output is critical for evaluating future change in patterns of hazard. The framework for indexing heatwaves by comparing a time-evolving measure of heat against some seasonally-varying percentile threshold is well-established in the literature (Baldwin et al. (2019); Schoetter et al. (2015); Acero et al. (2024); Argüeso et al. (2016)). Metrics such as heatwave frequency and duration are commonly used in hazard assessments, but there are few centralized tools and no universal heatwave criteria for computing them. This has resulted in parameter heterogeneity across the literature and has prompted some studies to adopt multiple definitions in an effort to build robustness (Sarah E. Perkins (2015)). However, many studies rely on only a handful of metrics and definitions due to the excessive data management and computational burden of sampling a greater number of parameters (S. E. Perkins & Alexander (2013)). The introduction of higher resolution global climate models and large ensembles has further complicated the development of software tools, which have remained mostly specific to individual studies and specific high performance computing systems. Some generalized tools have been developed to address this problem, but do not contain explicit methods for evaluating the potential sensitivities of heatwave hazard to the choices of heat measure, extreme heat threshold, and heatwave definition.

Development of the HDP was started in 2023 primarily to address the computational obstacles around handling terabyte-scale large ensembles, but quickly evolved to also investigate new scientific questions around how the selection of characteristic heatwave parameters may impact subsequent hazard analysis. By enabling the user to explicitly sample a large combination of parameters, the HDP can provide insight into how the spatio-temporal response of heatwaves to climate perturbations and forcings depends on the choice of heatwave parameters (e.g. [heatwave3](#), [xclim](#), [ehfheatwaves](#)).

40 **Key Features**

41 **Extension of XArray with Implementations of Dask and Numba**

42 xarray is a popular Python package used for geospatial analysis and for working with the
43 netCDF files produced by climate models. The HDP workflow is based around xarray and
44 seamlessly integrates with the xarray.DataArray data structure. By utilizing this well-adopted
45 framework, we increase the ease of use and portability of this package. Parallelization of HDP
46 functions is achieved through the integration of dask with automated chunking and task-
47 graph construction features that are built into the xarray library. Calculations are computed
48 per-grid-cell and compatible with any spatial configuration so long as it is defined by some
49 latitude and longitude.

50 The boost in computational performance the HDP offers over other heatwave diagnostic tools
51 comes from the combination of dask and numba. The dask Python package provides an
52 interface through which xarray.DataArray chunks are assigned to task-graphs and then be
53 dispatched across a cluster. The dask library handles many different job-dispatchers and can
54 conform to many different types of distributed-computing systems. This ensures the HDP can
55 be used on a variety of high performance computers and supercomputing clusters.

56 The numba Python package converts pure Python code and numpy function calls into compiled
57 machine code which can be executed much more quickly than the standard Python interpreter.
58 By writing the core heatwave-indexing and heatwave metric algorithms in Python and using
59 numba to convert them to machine code, we preserve the readability of the Python syntax
60 while dramatically increasing the computational efficiency of these algorithms both in terms
61 of speed and memory overhead. We then pass these numba-compiled functions to the dask
62 cluster for execution in parallel to leverage these improvements at scale.

63 **Heatwave Metrics for Multiple Measures, Thresholds, and Definitions**

64 The “heatwave parameter space” refers to the span of measures, thresholds, and definitions
65 that define individual heatwave “types.”

Parameter	Description	Example
Measure	The daily variable used to quantify heat.	Average temperature, minimum temperature, maximum temperature, heat index, etc.
Threshold	The minimum value of heat measure that indicates a “hot day.” This can be a fixed value or a percentile derived from a baseline dataset. The threshold can be constant or change relative to the day of year and/or location.	90th percentile temperature for each day of the year derived from observed temperatures from 1961 to 1990.
Definition	The pattern of hot days that constitutes a heatwave, described as a three number code.	“3-0-0” (three day heatwaves), “3-1-1” (three day heatwaves with possible one day breaks)

66 Heatwave studies are often based on a limited selection of these parameters (often only one
67 threshold and definition are used). The HDP allows the user to test a range of parameter
68 values: heatwaves that exceed 90th, 91st, ... 99th percentile thresholds for 3-day, 4-day, ...
69 7-day heatwaves. The multidimensional output produced by this sampling is elegantly stored

70 in `xarray.DataArray` structures that can be indexed and sliced for further analysis. Four
 71 heatwave metrics that evaluate the temporal patterns in each grid cell are calculated for each
 72 measure and aggregated into an `xarray.Dataset`.

Metric	Long Name	Units	Description
HWF	heatwave frequency	days	The number of heatwave days per heatwave season.
HWN	heatwave number	events	The number of heatwaves per heatwave season.
HWA	heatwave average	days	The average length of heatwaves per heatwave season.
HWD	heatwave duration	days	The length of the longest heatwave per heatwave season.

73 Diagnostic Notebooks and Figures

74 In addition to datasets which can be saved to disk, the HDP includes plotting functions
 75 and figure decks that summarize various metric diagnostics. These diagnostic plots are
 76 designed to give quick insight into potential differences in metric patterns between heatwave
 77 parameters. All figure-generating functions return instances of the `matplotlib.figure.Figure`
 78 class, allowing the user to modify the attributes and features of the existing plot or add
 79 additional features. These functions are contained within the `hdp.graphics` module which can
 80 be executed automatically through the full HDP workflow or imported by the user to create
 81 custom workflows.

82 The automatic workflow compiles a “figure deck” containing diagnostic plots for multiple
 83 heatwave parameters and input variables. The resulting deck may contain dozens of figures
 84 that can be difficult to parse through individually. To simplify this process, figure decks are
 85 serialized and stored in a single Jupyter Notebook that is separated into descriptive sections.
 86 This allows the user to keep all diagnostic figures in a single Notebook file and navigate through
 87 the plots using the Notebook interface. Markdown cells are added to the top of each figure
 88 that includes a basic description of the plotting function called and the variables used. The
 89 `HDPNotebook` class in `hdp.graphics.notebook` is utilized to facilitate the generation of these
 90 Notebooks internally, but can be called through the API as well to build custom notebooks.
 91 Below is an example of what a Notebook of the standard figure deck looks like:

Heatwave Diagnostics Package (HDP) Standard Figure Deck

Webpage: <https://github.com/AgentOxygen/HDP>
Version: HDP 0.2.0
Generation Timestamp: 2025-02-13 21:28
Figures Generated: 37

Figures 1

Figure 1.2

4-panel plot of time-evolution of heatwave metrics across multiple measures. Spatially-weighted by latitude, mean taken spatially and over both percentiles and all definitions. If ensemble is used, the range of member values is shaded behind the mean line. Abbreviations include percentile (perc) and definition (def).

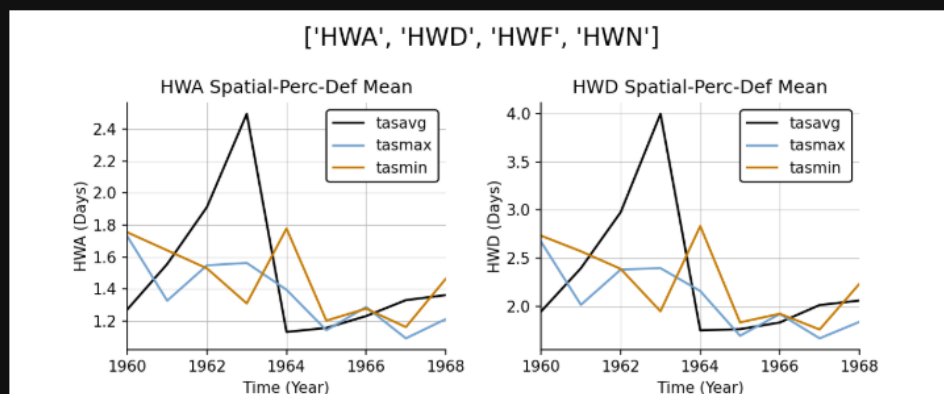


Figure 1: Example of an HDP standard figure deck

Ongoing Work

This package was used to produce the results featured in “Anthropogenic aerosol changes disproportionately impact the evolution of global heatwave hazard and exposure” by Dr. Geeta Persad, Cameron Cummins and Dr. Jane Baldwin, submitted to Environmental Research Letters in 2024. Updates to the HDP are ongoing and include, but are not limited to, adding new diagnostic plotting functions and developing heatwave metrics that measure spatial patterns. Additionally, we are planning to integrate this diagnostic package with the CESM Unified Post-Processing and Diagnostics suite (CUPiD) being developed by the National Center for Atmospheric Research.

Acknowledgements

We thank Dr. Tamas Loughran, Dr. Jane Baldwin, and Dr. Sarah Perkins-Kirkpatrick for their work on developing the initial Python software and heatwave analysis framework that inspired this project. Dr. Loughran’s Python package is available on [GitHub](#). This work is partially supported by the Modeling, Analysis, Predictions and Projections Award Program under the National Oceanic and Atmospheric Administration (Award Number NA23OAE4310601).

References

- Acero, F. J., Fernández-Fernández, M. I., Carrasco, V. M. S., Parey, S., Hoang, T. T. H., Dacunha-Castelle, D., & García, J. A. (2024). *Changes in heat waves characteristics over Extremadura (SW Spain)*. arXiv. <https://doi.org/10.48550/arXiv.2402.00514>
- Argüeso, D., Di Luca, A., Perkins-Kirkpatrick, S. E., & Evans, J. P. (2016). Seasonal mean temperature changes control future heat waves. *Geophysical Research Letters*, 43(14), 7653–7660. <https://doi.org/10.1002/2016GL069408>
- Baldwin, J. W., Dessy, J. B., Vecchi, G. A., & Oppenheimer, M. (2019). Temporally Compound Heat Wave Events and Global Warming: An Emerging Hazard. *Earth's Future*, 7(4), 411–427. <https://doi.org/10.1029/2018EF000989>
- Perkins, Sarah E. (2015). A review on the scientific understanding of heatwaves—Their measurement, driving mechanisms, and changes at the global scale. *Atmospheric Research*, 164–165, 242–267. <https://doi.org/10.1016/j.atmosres.2015.05.014>
- Perkins, S. E., & Alexander, L. V. (2013). *On the Measurement of Heat Waves*. <https://doi.org/10.1175/JCLI-D-12-00383.1>
- Schoetter, R., Cattiaux, J., & Douville, H. (2015). Changes of western European heat wave characteristics projected by the CMIP5 ensemble. *Climate Dynamics*, 45(5), 1601–1616. <https://doi.org/10.1007/s00382-014-2434-8>