Alarconpy: a Python Package for Meteorologists

Albenis Pérez-Alarcón[1] **and José Carlos Fernández-Alvarez**[1]

1 Department of Meteorology, Higher Institute of Technologies and Applied Sciences, University of Havana

Summary

In atmospheric science, researchers generally must process a large volume of data obtained from the network of meteorological observations on a local and global scale, as well as from the output of numerical weather forecast models. This data collection is stored in netCDF (Network Common Data Form) and GRIB (GRIdded Binary or General Regularly-distributed Information in Binary form, WMO (2003)) format. GRIB is a concise data format commonly used in meteorology to store forecast and historical meteorological data. It is standardized by the Commission for Basic Systems of the World Meteorological Organization (WMO).

These formats for storing meteorological data have led to the development of several applications for their processing, such as the NetCDF Operators (NCO) and the Climate Data Operators (CDO). The former focuses on simple data curation (e.g. viewing the contents of a file, selecting a subset of the data or editing the metadata within a file), while the latter provides for simple statistical analysis (e.g. calculating the climatology, percentile, correlation or heat wave index). It has also been observed in a rapid development of libraries in different programming languages such a Fortran, C, C++, MATLAB (MATrix LABoratory), Python, NCL (National Center for Atmospheric Research (NCAR) Command Language) and IDL (Interactive Data Language) to manage them.

Python is a modern, open-source, interpreted computer language whose use in the atmospheric and oceanic sciences is growing by leaps and bounds (Irving, 2019). In addition, Python offers the IPython interactive command line that allows scientists to view data, test new ideas, combine algorithmic approaches, and evaluate their results directly. This process could lead to an end result, or it could clarify how to build broader and more static production code (Pérez & Granger, 2007). Therefore, Python is an excellent tool for such a workflow (Yang et al., 1998). In fact, the development in Python of several libraries such as Metpy (R. M. May et al., 2020) and Siphon (R. May et al., 2017) have made it possible to integrate the functionalities of various standard Python packages in the processing of weather and oceanic data. The flexibility of Python as a programming language as well as the functions provided by various Python packages for the processing of weather data, motivated the development of the Alarconpy package.

Alarconpy is a set of tools developed in Python for the processing of meteorological data. It currently version is 1.0.3 and it includes functions to calculate some atmospheric variables such as relative humidity and the saturation vapor pressure, operations with dates and a module with specific algorithms for working with tropical cyclones (TCs). It also includes several predefined color palettes for plotting the wind field, the surface and mean sea level pressure, the radar reflectivity, the Gálvez-Davison Index (Gálvez & Davison, 2016), the precipitation as well as the color palettes available in Basemap (https://basemaptutorial.readthedocs.io/en/latest/).

Statement of Need

Alarconpy is used as the fundamental tool for processing the outputs of the numerical weather forecasting systems NTWF (Numerical Tools for Weather Forecast), NTHF (Numerical Tools for Hurricane Forecast) and HuMPI (Hurricame Maximum Potential Intensity) operating in the Department of Meteorology of the Higher Institute of Technologies and Applied Sciences, University of Havana. It integrates facilities that separately provide libraries such as Metpy (R. M. May et al., 2020), Scipy (https://www.scipy.org) and Cartopy (https://scitools.org.uk/cartopy/docs/latest/), as well as the development of new applications. In this way, the user is provided with a set of built-in tools for rapid weather data processing.

Acknowledgements

References

Gálvez, J. M., & Davison, M. (2016). *The Gálvez-Davison Index for tropical convection.* https://www.wpc.ncep.noaa.gov/international/gdi/GDI_Manuscript_V20161021.pdf

Irving, D. (2019). Python for atmosphere and ocean scientists. *Journal of Open Source Education*, *1*(11), 37. https://doi.org/10.21105/jose.00037

May, R., Arms, S., Leeman, J., & Chastang, J. (2017). *Siphon: A collection of Python utilities for accessing remote atmospheric and oceanic datasets* (Version 0.5.1). Unidata. https://doi.org/10.5065/D6CN72NW

May, R. M., Arms, S. C., Marsh, P., Bruning, E., Leeman, J. R., Goebbert, K., Thielen, J. E., & Bruick, Z. S. (2020). *MetPy: A Python package for meteorological data* (Version 0.12.1.post2). Unidata. https://doi.org/10.5065/D6WW7G29

Pérez, F., & Granger, B. E. (2007). IPython: A system for interactive scientific computing. *Computing in Science & Engineering*, *9*(3), 21–29. https://doi.org/10.1109/MCSE.2007.53

WMO. (2003). *General regularly-distributed information in binary form.* http://www.wmo.int/pages/prog/www/DPS/FM92-GRIB2-11-2003.pdf

Yang, T.-Y. B., Furnish, G., & Dubois, P. F. (1998). Steering object-oriented scientific computations. *Proc. Technology of Object-Oriented Languages and Systems (TOOLS), IEEE CS Press*, 112–119.