The Journal of Open Source Education

DIVAnd training: producing climatologies with Jupyter notebooks

Charles Troupin [1], **Alexander Barth** [1], and **Jean-Marie Beckers** [1]

1 GHER, Freshwater and Oceanic sCiences Unit of reSearch, University of Liège, Belgium

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Submitted: 28 January 2025
Published: unpublished

Summary

The DIVA-workshops project consists of a set of Jupyter notebooks, focused on creation of gridded fields from in situ observations using the `DIVAnd`. `DIVAnd` is a software tool, written in Julia, which perform interpolation in an arbitrary number of dimensions.

The notebooks address the different stages of the climatology generation: data reading and preparation, extraction of the topography and creation of a land-sea mask, setting of the spatial resolution and the time periods, estimation of the analysis parameters, analysis and creation of the metadata.

The target audience is wide as it includes: data analysts, who wish to create climatologies; physical oceanographers, who want to grid their observations for visualisation and potentially for quality control, programmers, who want to include the DIVAnd interpolation in a larger workflow involving other processing steps.

Statement of need

The gridding of in situ measurements is a common task in oceanography. It consists of the generation of one or several fields on a regular grid, using the information contained in a set of observations, generally sparsely distributed. The combination of such fields, produced at different depth levels and for different time periods, is often referred to as a climatology.

This gridding problem is not new and many methods have been developed during the last decades. One of the most widespread method is the Optimal Interpolation Bretherton et al. (1976), where analytical functions are used to specify the first guess error covariance. Since then, the method has been adapted and improved, yet it cannot easily address the decoupling of water masses separated a physical obstacle.

`DIVA` stands for Data-Interpolating Variational Analysis (Troupin et al., 2012) and is a analysis method based on the minimisation of a cost function. This cost function takes into account different constraint, typically the closeness to observations and the regularity (or smoothness) of the gridded field. DIVA, written in Fortran, was based on a finite-element solver and limited to two-dimensional applications. Climatologies were obtained by assembling 2D fields produced at specified depths and periods.

The Journal of Open Source Education

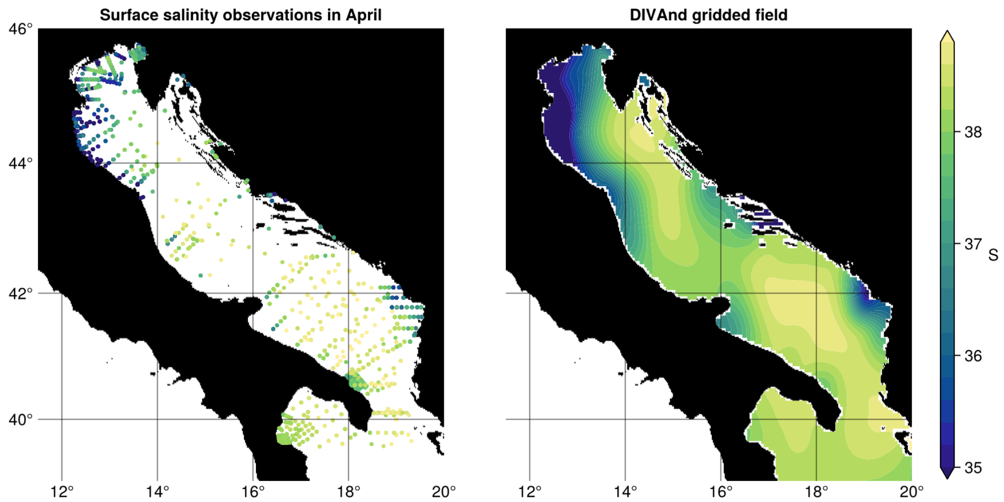

Figure 1: Example of salinity measurements and the corresponding analysed field.

`DIVAnd` (DIVA in n dimensions) is based on the same mathematical idea (the minimisation of a cost function) but extended to an arbitrary number of dimensions, typically longitude, latitude, depth and time (Barth et al., 2014). The code was first rewritten so that it can run on MATLAB and GNU Octave. Its performances were further improved thanks to the transition to the Julia language (Bezanson et al., 2017).

Without reviewing the full development history of the gridding and interpolation algorithms, we underline two specific aspects that are adequately addressed by DIVAnd (and DIVA) with respect to existing techniques: 1. The management of large datasets: the computation time is almost independent of the number of observations, making it possible to perform gridding with millions of data points. 2. The consideration of natural boundaries (coastlines, bottom topography) during the interpolation, hence avoiding the artificial mixing of water masses that are geographically close but separated by a physical obstacle.

The `DIVAnd` code is published on GitHub along with its documentation and examples, and the underlying theory has been published in Barth et al. (2014). However, in order to ensure that users are able to create their own climatologies, with a rather recent programming language, additional teaching resources were necessary. This is the main motivation behind the creation and the maintenance of the Diva-Workshops repository.

The DIVAnd learning module

The story

The first DIVA workshop was organised in Liège, Belgium, in 2006, in the frame of the European project Seadatanet (Schaap & Lowry, 2010). The goal was to teach users how to create climatologies by applying the DIVA (the two-dimensional, Fortran version) on their own dataset. Those training sessions were organised yearly until 2016 and allowed the creation of regional climatologies, published in the frame of European initiatives such as SeaDataNet or EMODnet (Martín Míguez et al., 2019).

Taking advantage of the Jupyter interface (Kluyver et al., 2016) and transition to Julia for the new version of DIVAnd, a set of notebooks was created as the main material for the user training. The first `DIVAnd` workshop took place in April 2018 in Liège. Since then, other training events were organised, while the training material is regularly used as the basis for the creation of gridded products for EMODnet Chemistry (Giorgetti et al., 2018). The choice of the Jupyter notebooks format was motivated by the interactivity and the step-by-step, documented approach.

The participant feedback is particularly valued, considering that it guide the development of new functionalities in the `DIVAnd` source code, but also the creation of new notebooks describing specific workflows (for instance the consideration of geostrophy) or the use of particular functions (for instance the use of an advection constrain in the interpolation).

Goal of the module

The goal of the training material module is twofold: 1. provide the users with a basic knowledge of Julia, meaning they are capable of reading the code presented in the notebooks, but also install new modules, write basic functions for processing or create basic plots. 2. endure that the users to able to create their own products (i.e. climatologies) by combining their own datasets with those from other sources (for instance the World Ocean Database) and setting the analysis parameters according to their region of interest.

Julia syntax bear similarities with other widespread languages, for instance MATLAB, yet some specificities have to explained to make sure users can make the most of it.

Instructional design

The notebooks have been organised by sub-folders according to their objectives: 1. Introduction: brief introduction to the Julia language and to the Jupyter notebooks, how to deal with netCDF files (reading and writing) and how to generate figures (maps, sections, …). 2. Preprocessing: preparation of the input required by `DIVAnd` (grid, time periods, bathymetry, observations) and estimation of the main analysis parameters (correlation length and noise-to-signal ratio). Code fragments dealing with various file formats (CSV, netCDF, TIFF, …) are also provided to help users work with the most frequent types of data. 3. Analysis: creating of gridded fields with DIVAnd, influence of the analysis parameters, and interpolation with different coordinate systems. 4. Advanced topics: this folder contains less frequently used notebooks, dealing with the generation of density maps, relative correlation length, background fields, advection constraints.

Since the notebooks require input data files (mainly bathymetry and observations) to be executed, we ensure those files are available from a public file server and downloaded locally whenever necessary.

Following our experience with users, for the creation of plots, the Makie module (Danisch & Krumbiegel, 2021) (along with `GeoMakie` for the maps) was selected to replce `PyPlot` (along with `Cartopy` (Met Office, 2010 - 2015) for the maps), which is based on the Python Matplotlib module (Hunter, 2007). Indeed, the import of PyPlot in the notebooks often generated errors on the user's machine, with sensitivity to the operating system and the pre-existing Python installation(s).

Users and applications

The user community mainly consists of scientists, data analysists and experts. This diversity implies that the content has to be taylored and sufficient to ensure users without any prior experience with Julia or Jupyter are able to run and modify the notebooks.

Among the applications, we can mention de EMODnet products (Webb et al., 2025). Other recent applications include the creation of climatologies and gridded fields for sea surface height (Doglioni et al., 2023) temperature and salinity (Shahzadi et al., 2021) and nutrients (Belgacem et al., 2021).

Acknowledgements

We acknowledge contributions from European Union's Horizon 2020 SeaDataCloud project (grant agreement No. 730960), from Horizon Europe research and innnovation FAIR-EASE project (grant agreement No. 101058785, DOI: 10.3030/101058785), Blue-Cloud 2026 (grant agreement No. 101094227, DOI: 10.3030/101094227) and IRISCC (grant agreement No. 101131261, DOI: 10.3030/101131261).

We wish to acknowledge the participants to the different editions of the DIVA workshops, since their feedback was essential to build and improve the content of the training sessions.

References

Barth, A., Beckers, J.-M., Troupin, C., Alvera-Azcárate, A., & Vandenbulcke, L. (2014). Divand-1.0: N-dimensional variational data analysis for ocean observations. *Geoscientific Model Development*, *7*(1), 225–241. https://doi.org/10.5194/gmd-7-225-2014

Belgacem, M., Schroeder, K., Barth, A., Troupin, C., Pavoni, B., Raimbault, P., Garcia, N., Borghini, M., & Chiggiato, J. (2021). Climatological distribution of dissolved inorganic nutrients in the western Mediterranean Sea (1981–2017). *Earth System Science Data*, *13*(12), 5915–5949. https://doi.org/10.5194/essd-13-5915-2021

Bezanson, J., Edelman, A., Karpinski, S., & Shah, V. B. (2017). Julia: A fresh approach to numerical computing. *SIAM Review*, *59*(1), 65–98. https://doi.org/10.1137/141000671

Bretherton, F. P., Davis, R. E., & Fandry, C. B. (1976). A technique for objective analysis and design of oceanographic experiments applied to MODE-73. *Deep Sea Research and Oceanographic Abstracts*, *23*(7), 559–582. https://doi.org/10.1016/0011-7471(76)90001-2

Danisch, S., & Krumbiegel, J. (2021). Makie.jl: Flexible high-performance data visualization for julia. *Journal of Open Source Software*, *6*(65), 3349. https://doi.org/10.21105/joss.03349

Doglioni, F., Ricker, R., Rabe, B., Barth, A., Troupin, C., & Kanzow, T. (2023). Sea surface height anomaly and geostrophic current velocity from altimetry measurements over the arctic ocean (2011–2020). *Earth System Science Data*, *15*(1), 225–263. https://doi.org/10.5194/essd-15-225-2023

Gandin, L. S. (1965). Objective analysis of meteorological fields. *Israel Program for Scientific Translation, Jerusalem*, 242 pp.

Giorgetti, A., Partescano, E., Barth, A., Buga, L., Gatti, J., Giorgi, G., Iona, A., Lipizer, M., Holdsworth, N., Larsen, M. M., Schaap, D., Vinci, M., & Wenzer, M. (2018). EMODnet chemistry spatial data infrastructure for marine observations and related information. *Ocean &Amp; Coastal Management*, *166*, 9–17. https://doi.org/10.1016/j.ocecoaman.2018.03.016

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, *9*(3), 90–95. https://doi.org/10.1109/MCSE.2007.55

Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B., Bussonnier, M., Frederic, J., Kelley, K., Hamrick, J., Grout, J., Corlay, S., Ivanov, P., Avila, D., Abdalla, S., Willing, C., & team, J. development. (2016). Jupyter notebooks - a publishing format for reproducible computational workflows. In F. Loizides & B. Scmidt (Eds.), *Positioning and power in academic publishing: Players, agents and agendas* (pp. 87–90). IOS Press. https://doi.org/10.3233/978-1-61499-649-1-87

Martín Míguez, B., Novellino, A., Vinci, M., Claus, S., Calewaert, J.-B., Vallius, H.,

The Journal of Open Source Education

Schmitt, T., Pititto, A., Giorgetti, A., Askew, N., Iona, S., Schaap, D., Pinardi, N., Harpham, Q., Kater, B. J., Populus, J., She, J., Palazov, A. V., McMeel, O., … Hernandez, F. (2019). The european marine observation and data network (EMODnet): Visions and roles of the gateway to marine data in europe. *Frontiers in Marine Science*, *6*. https://doi.org/10.3389/fmars.2019.00313

Met Office. (2010 - 2015). *Cartopy: A cartographic python library with a matplotlib interface*. https://doi.org/10.5281/zenodo.1182735

Schaap, D. M. A., & Lowry, R. K. (2010). SeaDataNet – pan-european infrastructure for marine and ocean data management: Unified access to distributed data sets. *International Journal of Digital Earth*, *3*(sup1), 50–69. https://doi.org/10.1080/17538941003660974

Shahzadi, K., Pinardi, N., Barth, A., Troupin, C., Lyubartsev, V., & Simoncelli, S. (2021). A new global ocean climatology. *Frontiers in Environmental Science*, *9*. https://doi.org/10.3389/fenvs.2021.711363

Troupin, C., Barth, A., Sirjacobs, D., Ouberdous, M., Brankart, J.-M., Brasseur, P., Rixen, M., Alvera-Azcárate, A., Belounis, M., Capet, A., Lenartz, F., Toussaint, M.-E., & Beckers, J.-M. (2012). Generation of analysis and consistent error fields using the data interpolating variational analysis (DIVA). *Ocean Modelling*, *52-53*, 90–101. https://doi.org/10.1016/j.ocemod.2012.05.002

Webb, T. J., Beja, J., Bejarano, S. J. F., Ramos, E., Sainz-Villegas, S., Soetaert, K., Stolte, W., Troupin, C., & Weigel, B. (2025). Realising the potential of interoperable data products to improve the outlook for marine biodiversity: Lessons from the european marine observation and data network. *Marine Policy*, *173*, 106578. https://doi.org/10.1016/j.marpol.2024.106578