Python Module Functions To Evaluate Science Data from Ocean Observatories Initiative (OOI) Gliders

Leila (Belabbassi) BaghdadBrahim | Rutgers University

Co-Authors

Lori Garzio | Rutgers University and Sage Lichtenwalner | Rutgers University

Abstract (Poster)

This work is part of the Ocean Observatories Initiative Science and Education (OOISE) project. The project goal is to identify and document data issues so that researchers and educators have a better understanding about OOI data access and use. For this work, data are from gliders deployed at any of the six OOI arrays: Pioneer, Endurance, Station Papa, Irminger Sea, Argentine Basin and Southern Ocean.

The objective of this work is to showcase a collection of python module functions developed to review and evaluate science data from the OOI gliders. The module functions are collected in a GitHub repository (ooi-data-lab) available to researchers and educators for general use.

Jupyter notebooks are used as a web-based interactive environment to share Python code, execution and visualization, to facilitate a demonstration of the process of data review and evaluation of the OOISE science data parameters from gliders. The workflow used for this work starts with data download and ends with data quality report generation. The automated data quality reports, combined with a suite of data plots, are subsequently used to (1) flag datasets that qualify for education activities and (2) add notes regarding data issues that are transmitted to the OOI project for further investigation and resolution. The module functions used to create the last two steps are not addressed in the Jupiter notebooks.

Step-One (Data Access): Demonstrate how to request and download glider data using the Machine-to-Machine (M2M) interface available for the open Application Program Interface (API) of the OOINet (https://ooinet.oceanobservatories.org/). Many features of the OOI API services are unraveled using pieces of codes to show how to request sensors, asset and events information. This is a crucial step that is often a substantial hurdle for many OOI users.

Step-Two (Data Analysis): Demonstrate how to use the automated module functions built to analyze the netCDF data files downloaded in Step-One and generate a set of quick look data quality reports. The reports are generated at the deployment and instrument level, and summarize results from 12 automated quality control tests that can be grouped in four categories (1) Deployments location differences, (2) Data coverage and gap identification, (3) Dataset selection and comparison (4) Data validation and missing parameters identification. A selection of code, execution and visualization is used to walk through each category.