Documentation EOSC-Future Ocean Indicators for the Dashboard of the state of the environment

There is a notebook in development that calculates indicators for the Ocean Variables Temperature, Oxygen, pH, Nitrate + Nitrite, Silicate, Phosphate based solely on in-situ data. Below I have provided a summary of the different steps of the notebook. In the future we would like to expand the data input to the indicators also with other blue data infrastructures.

Input

Input parameters:

- Sea region → Example: Western Mediterranean (WMED)
- Parameter name → Example: temperature
- Upper boundary → Example: 0 m
- Bottom boundary → Example: 10 m
- Start date → Example: 2010
- End date → Example: 2015

Bounded box is created based on the chosen **sea region**, with minimum and maximum values for longitude and latitude. → Example: WMED = lon_min, lon_max, lat_min, lat_max = [-7, 17, 28, 49]

The desired unit and related dimension vocab, minimum value and maximum value are chosen based on **parameter name.** → Example: Temperature; desired unit = °C, dimension vocab = UPAA, minimum value = -2.5, maximum value = 40.

Beacon Query

A post request is performed with the following input parameters towards Beacon which queries the SeaDataNet CDI database:

- Parameter name → Example: temperature
- Dimension vocab → Example: UPAA
- Minimum value → Example: -2.5 °C
- Maximum value → Example: 40 °C
- Start date → Example: 2010-01-01T00:00:00
- End date → Example: 2015-12-31T23:59:59
- Upper boundary → Example: 0 m
- Bottom boundary → Example: 10 m
- Lon_min, lon_max, lat_min, lat_max → Example: [-7, 17, 28, 49]

The output in netCDF is transformed into a Pandas Dataframe.

As an example, below we see 980,000 temperature measurements from the SeaDataNet CDI database that are performed over a period of 6 years (2010-2015) in the top 10 m of the ocean within the bounded box of the Western Mediterranean Sea region.

	SDN_ID	TIME	LONGITUDE	LATITUDE	temperature	DEPTH
observations						
0	1389285	2.455347e+06	-2.218500	46.228670	15.0142	5.218
1	1389285	2.455347e+06	-2.218500	46.228670	15.0132	5.388
2	1389285	2.455347e+06	-2.218500	46.228670	15.0149	5.604
3	1389285	2.455347e+06	-2.218500	46.228670	15.0147	5.808
4	1389285	2.455347e+06	-2.218500	46.228670	15.0161	6.005
...	...	...	...	...	...	...
191458	2380239	2.457128e+06	12.320958	45.290441	14.8000	0.500
191459	2380239	2.457102e+06	12.320958	45.290441	10.6000	0.500
191460	2380466	2.457218e+06	12.316756	45.088701	28.3000	0.500
191461	2380466	2.457190e+06	12.316756	45.088701	26.1000	0.500
191462	2380466	2.457128e+06	12.316756	45.088701	19.3000	0.500

980831 rows × 6 columns

Data processing

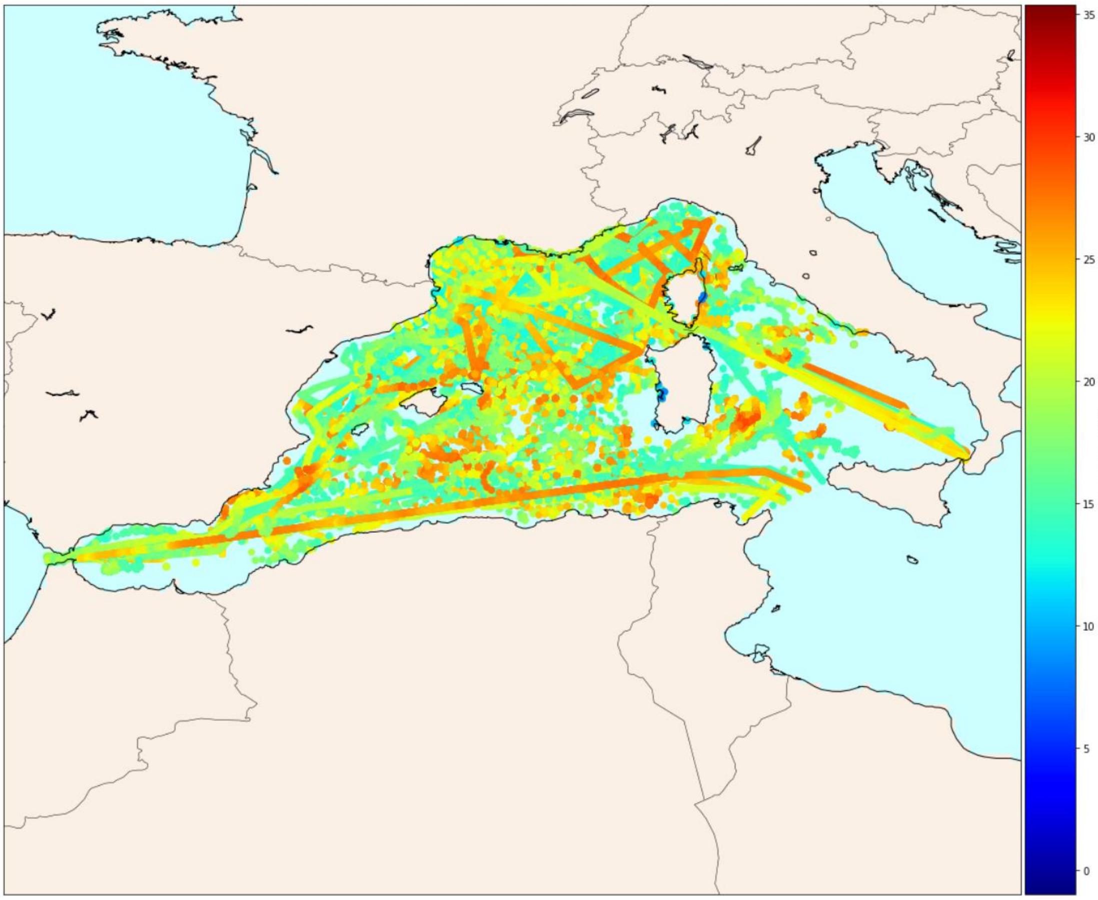
After the query is performed and data is obtained within the chosen bounded box, depth range and time period, a second filter is performed in order to do a more detailed area search. Use is made of shapefiles related to the sea regions to narrow down the data output such that the data lies inside the shapefile boundaries. The shapefile is transformed into a polygon with longitudes and latitudes. Then the Pandas Dataframe is converted into a GeoDataFrame and the data points are located that lie inside the polygon. As a result, in this example, we can see that 437,000 measurements remain.

	SDN_ID	TIME	LONGITUDE	LATITUDE	temperature	DEPTH	Season
Datetime							
2010-01-01 11:19:59.999982	153348	2.455198e+06	8.857	37.051	17.0900	6.00	Winter
2010-01-01 11:51:59.999988	153363	2.455198e+06	4.537	40.358	15.3740	8.00	Winter
2010-01-01 11:51:59.999988	153363	2.455198e+06	4.537	40.358	15.3700	4.00	Winter
2010-01-03 12:01:59.999992	153390	2.455200e+06	4.583	40.534	15.4670	8.00	Winter
2010-01-03 12:01:59.999992	153390	2.455200e+06	4.583	40.534	15.4640	4.00	Winter
...	...	...	...	...	...	...	...
2015-12-31 20:07:20.999995	1347667	2.457388e+06	0.850	38.743	16.8844	3.04	Autumn
2015-12-31 21:02:49.000008	455674	2.457388e+06	15.354	38.948	16.7530	3.90	Autumn
2015-12-31 21:02:49.000008	455674	2.457388e+06	15.354	38.948	16.7520	9.70	Autumn
2015-12-31 21:02:49.000008	455675	2.457388e+06	15.354	38.948	16.7520	9.70	Autumn
2015-12-31 21:02:49.000008	455675	2.457388e+06	15.354	38.948	16.7530	3.90	Autumn

437164 rows × 7 columns

The timestamps in Julian dates are converted to standard datetimes and sorted on date. For each data measurement it is determined in which season it was measured. For the seasons the Mediterranean seasons are taken as Winter being January, February, March, Spring being April, May, June, etc.

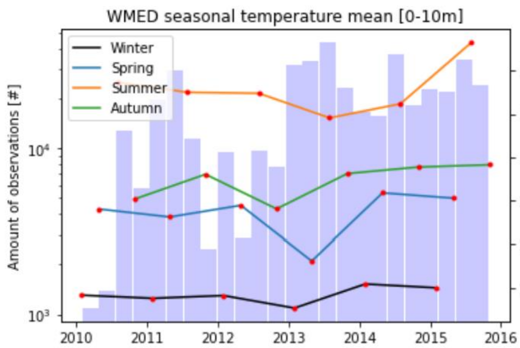
The data is then plotted on a map with accompanying colorbar to check the requested data.

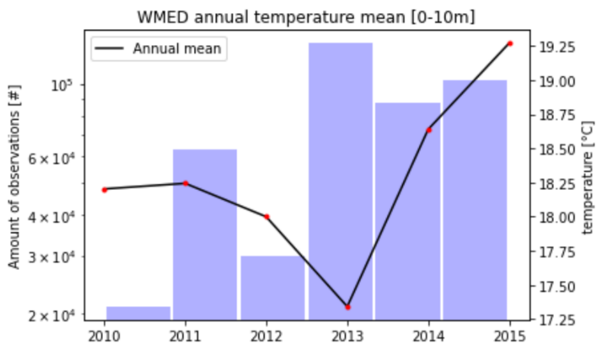


From these data measurements, the seasonal and annual means can be calculated. For the indicator "Seasonal Mean", for each year and season the mean is calculated as well as the number of observations. In order to compute the seasonal means, the Mediterranean seasons are defined as follows:

- Winter (month 1, 2, 3)
- Spring (month 4, 5, 6)
- Summer (month 7, 8, 9)
- Autumn (month 10, 11, 12)

You can see the seasonal and annual temperature means below.





For the indicator "Seasonal Mean standard deviation", for each year and season the standard deviations and means are calculated as well as the number of observations. For the indicator "Annual Mean standard deviation", for each year the standard deviations and means are calculated as well as the number of observations.