

# SHDR User Manual

Manuel Franco Pire

May 11, 2023

## Contents

<b>1</b>	<b>Description</b>	<b>2</b>
<b>2</b>	<b>Using SHDR</b>	<b>2</b>
2.1	Fit parameters . . . . .	3
2.2	Fitting a time series . . . . .	3
<b>3</b>	<b>Exploring SHDR results</b>	<b>4</b>
<b>A</b>	<b>Differential evolution algorithm</b>	<b>6</b>

# 1 Description

The SHDR algorithm (González-Pola et al., 2007) fits an upper ocean method for describing. This manual aims to be a guide for using the codes in the repository. For further, please refer to the original work by González-Pola et al.

SHDR stands for *Sharp Homogenization/Diffusive Retreat*. The thermocline is assumed as being a result of diffusive processes, with and the mixed layer depth (MLD) defining the boundary and the following analytical form can be found to describe an upper oceanic profile:

$$f(z) \begin{cases} a_1 & \text{if } z < D_1, \\ a_3 + b_3(z - D_1) + a_2 e^{(-b_2(z-D_1) - c_2(z-D_1)^2)} & \text{if } z > D_1. \end{cases} \quad (1)$$

The different parameters in equation 1 are presented graphically in figure 1, and can be interpreted as:

- $D_1$  - Mixed layer depth.
- $a_1$  - Temperature of the mixed layer depth.
- $b_2$  - Exponential decay coefficient at the seasonal thermocline.
- $c_2$  - Gaussian decay coefficient at the seasonal thermocline.
- $b_3$  - Gradient of the permanent thermocline.
- $a_2$  - Temperature difference between the mixed layer and the base of the thermocline.
- $a_3$  - Temperature at the base of the thermocline.

The optimization method used by SHDR is a differential evolution algorithm. To find the combination of parameters that minimize the mean squared error (MSE), a population of individuals is randomly generated. Each individual represents a certain combination of the parameters in equation 1. By means of crossing and mutations, the “genetics” of the individuals that perform best are favored, and the MSE is progressively minimized.

## 2 Using SHDR

The source code of SHDR is very simple. The code repository contains the main file `SHDR.py`. This file has a function `fit_profile`.

1. Download the `SHDR.py` file and place it in your working directory.
2. Using your preferred programming environment - python shell, jupyter notebook or python file - import the function `fit_profile` from the `SHDR.py` file.
3. Import your data.
4. Pass your data to the function `fit_time_series`, alongside with the options.

The `fit_profile` only fits a single profile. The profile is passed to the function as `y` and `z` coordinates.

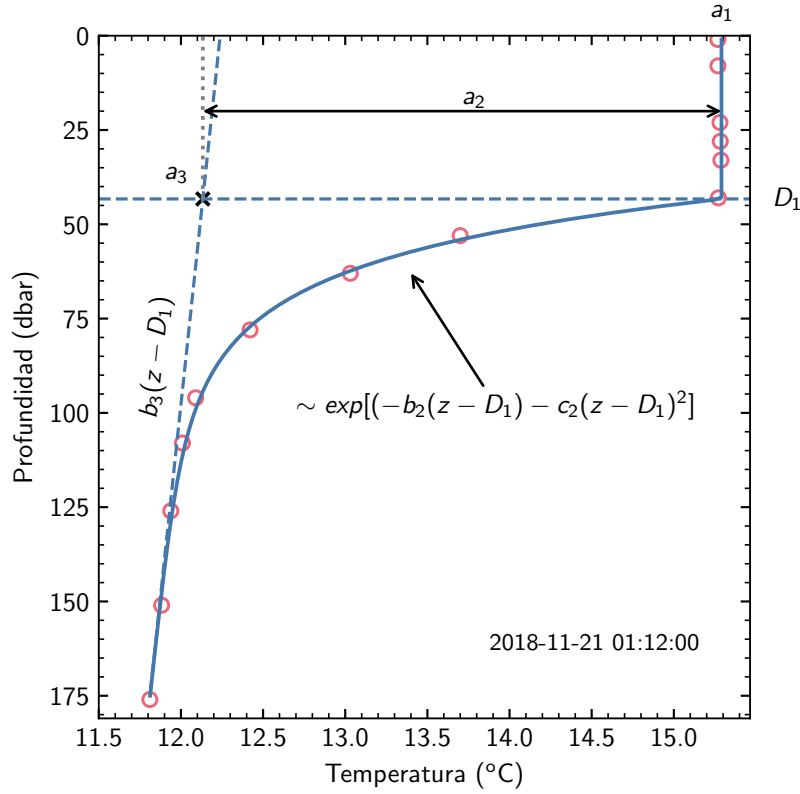


Figure 1: Caption

## 2.1 Fit parameters

The fitting algorithm has different parameters that can be passed as arguments to the `fit_profile` function. For most use cases,

- `max_depth`: Maximum depth in the profile to be considered for the fitting. Use this argument to specify the depth that you consider defines the upper ocean in your profiles. Default value is 450 dbar.
- `min_depth`:
- `min_obs`: Minimum number of points in the profile to perform the fit.

For a more advanced usage, the parameters of the differential evolution algorithm can be modified. The default values should reproduce consistent results for most CTD profiles:

- `CR` - Cross probability. Default: 0.7
- `FF` - Mutation factor. Default: 0.7
- `num_generations` - Maximum number of generations. Default: 1200
- `num_individuals` - Number of individuals in each generation. Default: 60
- `tol` - tolerance to stop evolution. Default: 0.00025

## 2.2 Fitting a time series

As many datasets are organized as time series, the **SHDR** module includes a function to fit a time series. The function is called `fit_time_series`. Its optional parameters are the

same as above. Its main parameters are:

- **time** - 1D Array defining the timestamps of the measurements
- **variable** - 2D Array containing the measurements. The first dimension of the array must be time, and the same length as **time**
- **depth** - 2D Array defining the depths of the measurements. The first dimension must be temporal.
- **lat** - 1D Array defining the latitude of each measurement.
- **lon** - 1D Array defining the longitude of each measurement.
- **save** - Defines a relative path to store the results as a .csv file. If this argument is not passed, the results won't be stored.

Only the parameters **variable** and **depth** as used to perform the fitting, the rest are required to index the results of the fit. **fit\_time\_series** returns a Pandas DataFrame containing the results of the fit.

### 3 Exploring SHDR results

## References

- González-Pola, C., Fernández-Díaz, J. M., & Lavín, A. (2007). Vertical structure of the upper ocean from profiles fitted to physically consistent functional forms. *Deep Sea Research Part I: Oceanographic Research Papers*, 54(11), 1985–2004. <https://doi.org/10.1016/j.dsr.2007.08.007>

## A Differential evolution algorithm

The differential evolution algorithm implemented in SHDR is as follows following. Firstly, an initial population is generated randomly following a uniform distribution inside the limits for each parameter. Each generation  $g$  evolves according to the following algorithm:

1. The MSE is computed for each individual, and the best is detected,  $\mathbf{x}_{b,g}$ .
2. For each individual  $\mathbf{x}_{i,g}$ , two individuals  $\mathbf{x}_{k_1,g}, \mathbf{x}_{k_2,g}$  are selected randomly with  $i \neq k_1 \neq k_2$ , and its mutation  $\mathbf{y}_i$  is computed:

$$\mathbf{y}_{i,g} = (1 - \mu)\mathbf{x}_{i,g} + \mu\mathbf{x}_{b,g} + F(\mathbf{x}_{k_1,g} - \mathbf{x}_{k_2,g}), \quad (2)$$

where  $FF \in (0, 1)$  is the mutation factor and  $\mu \in (0, 1)$  the weight of the best individual.

3. In the next generation, the individual will evolve from  $\mathbf{x}$  a  $\mathbf{y}$  following

$$\mathbf{x}_{i,g+1} = \begin{cases} \mathbf{y}_{i,g} & \text{if } r_i < CR, \\ \mathbf{x}_{i,g} & \text{otherwise,} \end{cases} \quad (3)$$

where  $CR \in (0, 1)$  is the cross probability  $r_i \sim U[0, 1)$ .

4. If a parameter of a new individual exceeds the limits, it is returned to the closest limit.
5. If  $g + 1 = g_{\max}$  with  $g_{\max}$  the maximum number of generation, or if the mean  $\overline{em} < \text{tol} \cdot \sigma_{em}$ , where  $e_m$  is the MSE,  $\sigma_{em}$  is the standard deviation of the population MSE, and  $\text{tol}$  is a tolerance previously defined, the iteration is halted and the best individual of the new generation  $g + 1$  is taken as solution. Otherwise, ..