COWBOE

Release 1.0.9

Naveen Vasudevan, Li Xi

Oct 02, 2023

CONTENTS:

CHAPTER

ONE

COWBOE PACKAGE

1.1 cowboe.cowboe module

COWBOE - Construction Of Windows Based On free Energy. Package for optimization and selection of parameters for umbrella sampling. project website: https://github.com/kuroonai/cowboe

cowboe.cowboe.**NMprogress**(***kwargs*)

provides an update on the progress of the algorithm by generating gif for steps and summarizies the function values in term of figures and convergence.

Parameters:

progessfile
[str] Name of the pogress file which hosts the parameter values in 3 columns A, V, fit() in groups of 3 (three vertices of the simplex at any given step)

Returns:
None

cowboe.cowboe.**cowboe**(***kwargs*)

cowboe algorithm for iteration and window selection

Parameters:

A = float
Optimization parameter 'A' for NM algorithm and parameter 1 of cowboe.

B = float
Parameter 'B' for the cowboe equation

V = float
Optimization parameter for NM algorithm which controls energy barrier.

sc = int
Sampling considered for each windows in conventional method in nano seconds e.g. 8 ns

name = str
Name of the point being evaluated

subtype = str
Name of the sub type of the system

location = string
Location of the pickled variable file created from the test file in pmftopoints().

Returns:
None

cowboe.cowboe.**cowboe3Dsurface**(***kwargs*)

Constructs a 3D surface plot based on the evaluvated unique parameter combinations and saves it in mp4 format. Need to have ffmpeg installed in path to use this function.

Parameters:

progessfile
[str] Name of the pogress file which hosts the parameter values in 3 columns A, V, fit() in groups of 3 (three vertices of the simplex at any given step)

fps
[int, optional] Frames per second for the animation. Default value is 15

dpi
[int, optional] Dots per inch value for the animation. Default value is 300

name
[str, optional] File name for the animation. Default value is 'cowboe3Dsurface.mp4'

Returns:
None

cowboe.cowboe.**cowboeKS**(***kwargs*)

Computes the Kolmogorov-Smirnov statistic on 2 samples. The null hypothesis is that the two individual samples were extracted from the same distribution and if the p-value is large or the KS statistics is small, then we cannot reject the hypothesis that the distributions of the two samples are the same.

Parameters:

location
[str] Location of all the trajectory files for a given point.

listfile
[str] Name of the file containing the list of all the trajectory files in the ascending order of windows. Can be same as the metadata file used for wham calculation or just a list of each window's trajectory file's names.

percentage
[float or str] Percentage of the total sampling to perform the test on eg. if 80 is given window n's total data will be compared to 80% of its data to test the null hypothesis.

Returns:
None

cowboe.cowboe.**cowboeNM**(***kwargs*)

Nelder-Mead optimization algorithm for the cowboe module.

Parameters:

A = array
A values of the 3 initial points for the 2 parameter optimization.

V = array
V or energy barrier values of the 3 initial points for the 2 parameter optimization.

fit = array
Fitness or the area difference value between the benchmark and the test case.

Returns:

conv = dict.
Dictionary with possible moves for the current simplex

cowboe.cowboe.**cowboeRNM**(***kwargs*)

(Restricted) Nelder-Mead optimization algorithm for the cowboe module .

Parameters:

A = array
A values of the 3 initial points for the 2 parameter optimization.

V = array
V or energy barrier values of the 3 initial points for the 2 parameter optimization.

fit = array
Fitness or the area difference value between the benchmark and the test case.

Returns:

conv = dict.
Dictionary with possible moves for the current simplex

cowboe.cowboe.**cowboe_OVL**(***kwargs*)

Calculates the coefficient of overlap (OVL) for different window's distribution

Parameters:

location
[str] Location of all the trajectory files for a given point.

listfile
[str] Name of the file containing the list of all the trajectory files in the ascending order of windows. Can be same as the metadata file used for wham calculation or just a list of each window's trajectory file's names.

name
[str] Name for the Overlap calculation (generally name of the point).

distplot
[bool] Switches the distribution plot with overlap on or off

Returns:
None

cowboe.cowboe.**cowboe_pmfplot**(***kwargs*)

Plots the pmf curve of a given free energy file.

Parameters:

pmf
[str] Name of the curve 1 (PMF) file.

name
[str] Name to save the output with.

Returns:
None

cowboe.cowboe.**cowboe_trajcut**(***kwargs*)

Slices the trajectories and creates new trajectory files with given percentage of the total sampling.

Parameters:

start
[int] Index value of the starting point from where the percentage of the data will be extracted. to start from the beginnning just provide "0" (without the quotes).

percentage
[float] Percentage of the sampling (points) to use from total sampling.

location
[str] Location of all the trajectory files for a given point.

listfile
[str] Name of the file containing the list of all the trajectory files in the ascending order of windows. Can be same as the metadata file used for wham calculation or just a list of each window's trajectory file's names (with extension).

name
[str] Name of the point.

Returns:
None

cowboe.cowboe.**cowboe_wham**(***kwargs*)

WHAM wrapper for PMF generation using trajectory files and user must have wham installed in system path

Grossfield, Alan, "WHAM: the weighted histogram analysis method", version 2.0.10, http://membrane.urmc.rochester.edu/wordpress/?page_id=126

Parameters:

name
[str] Name for the free energy file.

location
[str] location of the folder with the trajectory files

MCtrials
[int] Number of Monte Carlo trails (for bootstrapping). set to 0 if no bootstrapping error analysis is required.

hist_min: float
Minimum value for the histogram

hist_max
[float] Maximum value for the histogram

num_bins
[int] Total number of bins

tol
[float] Tolerance for the decimal places

temp
[float] Temperature at which simulation is done, can be assigned through metadatafile entries

numpad
[int] Numpad value for wham calculation, 0 for periodic PMF

metadatafile
[string] Name of the metadata file

Returns:
None

cowboe.cowboe.**cowboefit**(***kwargs*)

Finds the fitness or area difference between the test and benchmark pmf curves

Parameters:

test
[str] Name of the pmf curve being tested (point name used in pmftopoints()).

bench
[str] Name of the benchmarck pmf curve

frommin
[bool] Specify whether to consider error only after the minimum positions i.e. from x where pmf=0

Returns:

outdict
[dict] Dictionary with all the calculated deviation information.

cowboe.cowboe.**pmfcompare**(***kwargs*)

Plots the error bars of the two curves

Parameters:

pmfs
[list or tuple] Names of the PMF files to be plotted.

name
[str] Name to save the output with.

splice
[array] Array for index value for each pmf curve to splice to

markzero
[Bool] Whether to mark y = 0 with a dashed line or not

markers
[list] List of matplotlib markers to use for each curve

colors
[list] List of matplotlib plot colors to use for each curve

linestyles
[list] List of matplotlib plot line styles to use for each curve

mfc
[str] Takes input for the marker face color

lloc
[str] Legend location 'inside' or 'outside'

Returns:
None

cowboe.cowboe.**pmfdiff**(***kwargs*)

Plots two pmf curves and the abs difference between them.

Parameters:

pmf1
[str] Name of the curve 1 (PMF) file.

pmf2
[str] Name of the curve 2 (PMF) file.

name
[str] Name to save the output with.

Returns:
None

cowboe.cowboe.**pmftopoints**(***kwargs*)

Takes the test pmf file as input and generates gradient and initial guess for windows

Parameters:

location
[string] Location to save the pickled varible file created from the test file.

testpmf
[string] Name of the test pmf file.

order
[int] Order for polynomial fit.

Returns:
None

cowboe.cowboe.**progressfile**(***kwargs*)

To create the progress file from an n*3*3 array. Gives the progress.txt file.

Parameters:

points
[numpy array in the shape of n*3*3] An array of list of all the simplexes obtained.

Returns:
None

cowboe.cowboe.**settings_update**()

Prints instruction for updating default settings for the cowboe module and the wham calculation.

Returns:
None

CHAPTER

TWO

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

INDEX