README (v1.0)

Step 1: Create Input Folder. If you want to construct a new stack from a set of records, create a folder with a user specified name in Inputs/Stack_Construction/. If you want to align a set of records to a given stack, create a folder in Inputs/Core_Alignments/ instead.

Step 2: Define Run Settings. The folder created in the last step needs to contain 1-5 text files. If optional settings are omitted, the default setting is used.

2.1. For the stack construction,

- 'records.txt' contains a list of cores to be used.
- (Optional) Define all run specific settings in 'setting_stack.txt'. If this file is not in the folder, the algorithm uses 'Defaults/setting_stack.txt'. The following seven items are defined in the 'setting_stack.txt' file:
    - 'data_type' indicates whether the algorithm constructs the stack using $\delta^{18}O$ only (d18O) or both $\delta^{18}O$ and ^{14}C (both) in inferring ages. The default is 'both'.
    - 'kernel_function' indicates the kernel function used in the regression. There are four options: squared-exponential (SE), Matérn 3/2, 5/2 (M15 and M25), and Ornstein-Uhlenbeck (OU). The default is 'OU'.
    - 'IsLearn_transition' indicates whether the transition parameters are learned globally (global), locally (local), or not to be learned (no). The default is 'no'.
    - 'nSamples' indicates the number of sample paths in learning parameters and the stack. The default is '100'.
    - 'start_age' indicates the start year of the stack in kiloyears. The start age is set to be the maximum value of this and that defined in the initial stack. The default is '-inf'.
    - 'end_age' indicates the end year of the stack in kiloyears. The end age is set to be the minimum value of this and that defined in the initial stack. The default is 'inf'.
    - 'interval' indicates the length of interval in which the ages in the stack are defined in kiloyears. The default is '0.1'. For example, if this value is set to be 0.1, then the resulted stack is defined every 0.1 kiloyear. Our model can compute the mean and standard deviation at every query age, however, run time will increase with query age resolution. In practice, it is best to compute all means and standard deviations on a set of finely arranged ages and linearly interpolate to calculate additional ages.
- (Optional) Define the initial stack in 'initial_stack.txt'. It should contain three columns: age, mean, and standard deviation. The first row consists of column titles. If 'initial_stack.txt' is not in the input folder, the algorithm will use 'Defaults/stack.txt'. The default stack is the global stack presented in Lisiecki & Stern (2016) from 0 to 150 ka BP. We also include eight regional stacks from Lisiecki & Stern (2016) as initial stack options. To use the regional stacks as the initial stack, state the appropriate acronym in the 'initial_stack.txt' file, instead of the 'age', 'mean', and 'std' columns. We suggest the user choose an initial stack based on the core locations. For example, if the user wishes to construct a stack from a population of mid-depth cores in the north Atlantic, than the user should use the Intermediate North Atlantic (INA) stack for the initial stack (see Table S2, Lisiecki & Stern, 2016). Additional stacks can be added to the default folder by storing the stack text file in 'Defaults/Regional_Stacks' folder. The user can use this stack by entering the stack title in the 'initial_stack.txt' file. The following are the regional stacks presented in Lisiecki & Stern (2016).

- INA: Intermediate North Atlantic Stack
- ISA: Intermediate South Atlantic Stack.
- DNA: Deep North Atlantic Stack.
- DSA: Deep South Atlantic Stack.
- IP: Intermediate Pacific Stack.
- DP: Deep Pacific Stack.
- II: Intermediate Indian Stack.
- DI: Deep Indian Stack.

- (Optional) Define the initial transition parameters in 'transition_parameter.txt'. We strongly advise the user not change this parameter. It is supposed to contain a 4 by 3 array, where the first row indicates the initial probability of contraction, steady, and expansion, the second row the transition probability of contraction to contraction, steady, and expansion, the third that of steady to contraction, steady, and expansion, and the fourth that of expansion to contraction, steady, and expansion. If it is not in the folder, the algorithm automatically uses 'Defaults/transition_parameter.txt' in the initialization.
- (Optional) Define the hyperparameters in 'hyperparameters.txt'. If it is not in the folder, the algorithm automatically uses 'Defaults/hyperparameters.txt' instead. It has the following six items to be defined:
  - 'alpha' and 'beta' indicates shape and rate parameters of the gamma distribution used in the transition model, respectively.
  - 'q' indicates the prior probability of being an outlier.
  - 'd' indicates how much outliers are deviated from the inliers.
  - 'nParticles' indicates the number of particles in the particle smoothing algorithm.
  - 'v' indicates the width of proposal distributions in the particle smoothing algorithm.
  - 'max_iters' indicates the maximum number of iterations in the MCMC algorithm.
  - 'h' is the tuning hyperparameter (bandwidth) in learning average sedimentation rates, which will be used only if 'islearn_average_sed_rate' is set to be 'adaptive'.

2.2. For the record alignments:

- Write down a list of records to be aligned in 'records.txt'.
- (Optional) Define all settings in 'setting_align.txt'. If this file is not in the input folder, the algorithm uses 'Defaults/setting_align.txt'. The following items are defined in 'setting_align.txt':
  - 'data_type' indicates whether the algorithm uses $\delta^{18}O$ only (d18O), ^{14}C only (C14) or both $\delta^{18}O$ and ^{14}C (both) in inferring ages. The default is 'both'.
  - 'IsLearn_transition' indicates whether the transition parameters are learned globally (global), locally (local), or not to be learned (no). The default is 'no'.
  - 'nSamples_learning' indicates the number of sample paths in learning parameters. The default is '100'.
  - 'nSamples_drawing' indicates the number of sample paths drawn after learning parameters as the result. The default is '1000'. Run time will increase with 'nSamples_drawing'.

- (Optional) Define the target stack in 'stack.txt'. For instruction on how to assign this setting, see the 'initial_stack.txt' explanation in 2.1.
- (Optional) Define the initial transition parameters in 'transition_parameter.txt'. It is supposed to contain 4 times 3 array, where the first row indicates the initial probability of contraction, steady, and expansion, the second the transition probability of contraction to contraction, steady, and expansion, the third that of steady to contraction, steady, and expansion, and the fourth that of expansion to contraction, steady, and expansion. If it is not in the folder, the algorithm automatically uses 'Defaults/transition_parameter.txt' in the initialization.
- (Optional) Define the hyperparameters in 'hyperparameters.txt'. If it is not in the folder, the algorithm automatically uses 'Defaults/hyperparameters.txt' instead. It has the following six items to be defined:
    - 'alpha' and 'beta' indicates shape and rate parameters of the gamma distribution used in the transition model, respectively.
    - 'q' indicates the prior probability of being an outlier.
    - 'd' indicates how much outliers are deviated from the inliers.
    - 'nParticles' indicates the number of particles in the particle smoothing algorithm.
    - 'v' indicates the width of proposal distributions in the particle smoothing algorithm.
    - 'max_iters' indicates the maximum number of iterations in the MCMC algorithm.
    - 'h' is the tuning hyperparameter (bandwidth) in learning average sedimentation rates, which will be used only if 'islearn_average_sed_rate' is set to be 'adaptive'.

Step 3: Define Core Settings. A folder for each core should be placed in the folder 'Cores'. Each core folder should contain some of the following four text files dependent on which 'data_type' is being used.

- d18O_data.txt has two-columns: depths (in meters) and $\delta^{18}O$ observations (in per mil). Observations at one depth are accepted. In fact, more observations at the same depth would make its age inference more accurate. If only ^{14}C data is used, this file can be omitted.
- C14_data.txt has six columns. The first column is depths (in meters), the second is uncalibrated ^{14}C years (in kiloyears), the third is errors (in kiloyears), the is fourth reservoir ages (in kiloyears), the fifth is standard deviations of reservoir age uncertainty (in kiloyears), and the sixth is the type of calibration curve (1: IntCal13, 2: Marine13, 3: SHCal13). Observations at one depth are accepted. If only $\delta^{18}O$ data is used, this file can be omitted.
- additional_ages.txt has four-column values. The first is depths (in meters), the second is age guesses (in kiloyears), the third is errors (in kiloyears), and the fourth is modes. Two different modes can be specified. If mode is set to 0 the age guess is assumed to follow a Gaussian distribution with the age guess as its mean and error as its standard deviation. If mode is set to 1 the age guess is assumed to be an interval where its boundaries are the age guess ± error. One depth must have at most one set of values. If 'data_type' is set to be 'C14', this file will be ignored and can be omitted. If no such information is available, this file can be omitted. If the 'data_type' is set to d18O only, we recommend specifying a start and end age to assist the alignment procedure.
- (Optional) setting_core.txt has all record-specific settings for inferring ages. If it is not in the folder, the algorithm uses 'Defaults/setting_core.txt'.
    - 'initial_shift' defines an initial guess on the shift parameter. If no such a guess exists, leave it 'NaN', which will be initialized in the procedure: default is 'NaN'.

- ‘initial_scale’ define an initial guess on the scale parameter. If not such a guess exists, leave it ‘NaN’, which will be initialized in the procedure: default is ‘1’.
- ‘initial_average_sed_rate’ defines an initial guess on the average sedimentation rate (m/ky). If not such a guess exists, leave it ‘NaN’, which will be initialized in the procedure: default is ‘NaN’.
- ‘islearn_shift’ determines if the shift parameter is learned in the procedure (‘yes’ or ‘no’): default is ‘yes’.
- ‘islearn_scale’ determines if the scale parameter is learned in the procedure (‘yes’ or ‘no’): default is ‘no’.
- ‘islearn_average_sed_rate’ determines how the average sedimentation rate is learned in the procedure. If it is considered as a constant parameter, set it ‘constant’; if it is also supposed to be learned as a (very smooth) function over ages, set it ‘adaptive’; if you do not want to learn it, set if ‘no’. The default is ‘adaptive’.
- ‘lower_bound’ defines the lower bound of inferred ages. If no estimate exists, leave‘NaN’ and ‘lower_bound’ will be set by the lowest age in the stack if the ‘data_type’ is set to $\delta^{18}O$ only or both, and 0 if the ‘data_type’ is set to ^{14}C only. The default value is ‘NaN’. Note that this setting does not give additional age information to the shallowest depth.
- ‘upper_bound’ defines the upper bound of inferred ages. If no estimate exists, leave ‘NaN’, which and ‘upper_bould’ will be set by the highest age in the stack if the ‘data_type’ is set to $\delta^{18}O$ only or both, and 60 if the ‘data_type’ is set to ^{14}C only. The default is ‘NaN’. Note that this setting does not give additional age information to the deepest depth.
- ‘min_resolution’ defines how many noninformative depths are additionally used: they are inserted as few as possible so that any adjacent depths are less than ‘min_resolution’ times the length of core. For example, if a core is 3 meters long and ‘min_resolution’ is set to be 0.1, then minimum number of additional depths are inserted so that no pair of adjacent depth is more than 30 centimeters. If no additional noninformative depth is needed, set it ‘NaN’. The default is ‘0.05’.
- ‘lower_sedrate’ defines the minimum value of rescaled (normalized) sedimentation rates. The default is ‘-inf’, which means that there is no such a constraint.
- ‘upper_sedrate’ defines the maximum value of rescaled (normalized) sedimentation rates. The default is ‘inf’, which means that there is no such a constraint.

- (Optional) ‘query_depths.txt’ has the settings required to interpolate the age inferences. If it is not in the folder, the algorithm will return the text files about confidence bands to the depths used in the inference.
  - ‘start_depth’ defines the start depth (in meters) of the query depths. If it is either not listed or defined as ‘NaN’, the algorithm will assign the start (shallowest) depth of the data to it.
  - ‘end_depth’ defines the end depth (in meters) of the query depths. If it is either not listed or defined as ‘NaN’, the algorithm will assign the end (deepest) depth of the data to it.
  - ‘interval’ defines the interval (in meters) of the query depths. If it is not listed, the algorithm will assign ‘0.01’ (i.e., 1 centimeter) to it.

Step 4: Run the Software. Execute the command addpath(‘Codes/’); and then results = Stack_Learner(inputFile); for stack construction or results = Core_Aligner(inputFile); for record alignments, where inputFile indicates the name (not path) of the folder made in step 1. All results are stored in the folder ‘Outputs/Stack_Construction/’ or ‘Outputs/Core_Alignments/’. Example codes are located in ‘Main_Stack_Construction.m’ and

'Main_Core_Alignments.m'. Results are consisting of the following 3-4 text files:

- results.mat contains all results.
- The summary of inferred ages is stored as a set of text files in /ages.
- stack.txt contains the constructed stack from the stack construction algorithm. It is not generated by the record alignment algorithm.

Step 5: View the Results. Run any of the following commands to visualize the results:

- AgeVsDepth(results_path); returns figures of age assignments to depths in each record in results_path. If you set multiple paths in results_path, then it will compare two sets of results. An example code can be found in 'Main_AgeVsDepth.m'.
- AgeVsD18O(results_path); returns figures of alignments of each record in results_path to the stack to be aligned: results_path is set to take only one path. If some depths have multiple $\delta^{18}O$ observations, then additional figure will be represented for them. An example code can be found in 'Main_AgeVsD18O.m'.
- AgeVsSedRate(results_path); returns figures of empirical sedimentation rates from sample paths with average sedimentation rate curves in results_path. If you set multiple paths in results_path, then it will compare two sets of results. It also returns rescaled log-sedimentation rates with their boundaries in another figures. An example code can be found in 'Main_AgeVsSedRate.m'.
- AgeVsAge(titles,results_path); returns figures of comparison between two age inferences in results_path, where their titles are defined in titles: results_path must take two paths. An example code can be found in 'Main_AgeVsAge.m'.
- Stack_Summary(results_path); returns several figures related to the results from a stack construction in results_path: results_path is set to take only one path. An example code can be found in 'Main_Stack_Summary.m'.
- Stack_Comparison(titles,results_path); returns figures of comparison between multiple stacks in results_path, where their titles are defined in titles. An example code can be found in 'Main_Stack_Comparison.m'.