Downscaling-User Guide

ekanaykk

July 2023

1 Overview

Downsacling will be performed in 3 steps. In our proposed downscaling framework, we first perform standard downscaling and model it jointly with observational data using multivariate Basis Graphical Lasso (BGL).

1. Preprocessing and Standard Downscaling: Here we first extract climate model data and high-resolution data from netcdf files into matrix format. Then we perform standard downscaling for the observational period and the for the future years we plan to perform the downscaling task.
2. BGL Downscaling: We fit the BGL models (separated by months) here using data for the observational period. Given the estimates from standard downscaling for future, fitted BGL model is used to produce high-resolution projections for future years. Downscaling for each future day is separately performed and saved into csv files.
3. Combine into netcdf files: Here, we finally combine all the downscaled days into years and create netcdf files. Each year will have a separate netcdf file containing all the locations around the globe.

2 Basic set-up

Each .sh file at line 16 load required modules. Please make sure you are able to load them before submitting the jobs.

Below R packages are required to perform downscaling. Please install the packages before starting the jobs.

- ncdf4
- data.table
- akima
- sp
- pryr

- foreach
- doParallel

To install R packages please use following commands.

1. module load /nasa/modulefiles/pkgsrc/toss4/pkgsrc/2022Q1-rome
2. export R_LIBS=~/R-packages
3. module load gcc/10.3
4. R
5. install.packages("Name of the package")
6. Type "1" as the selection of CRAN mirror.
7. Type "yes" to all the questions.
8. Type q() to quit R.
9. Type "n" when it asks to save the workspace.

3 Submitting jobs

There are seven variables which need to be specified at each step depending on the downscaling job.

1. model: This variable specifies the name of the climate model (eg. CMCC-ESM2_r1i1p1f1_gn)
2. scenario: This variable specifies the climate scenario (eg. ssp126)
3. GCM_variable: This variable specifies the name of the GCM variable (eg. tasmin)
4. era_variable: Name of the ERA variable (eg. Tair)
5. era_variable_spe: Specify whether minima or maxima of the ERA variable (eg. minima)
6. path0: Path to the main directory where the GCM model files and ERA files located. Also all resulting files will be saved within this main directory (in separated sub-directories). Change the path as necessary.
7. Downscale_years: Specify the years to downscale.

   NOTE: If you wish to save R output files in a separate location please modify the last line of each of the Preprocess.sh Downscaling.sh and NC.file.sh as necessary.

3.1 Preprocessing and Standard Downscaling

This step will extract GCM data and ERA data into .csv files in matrix forms. Separate model is fitted for each sub-region and each month. Therefore, all the files will be saved in separate region specific sub-directories and will be separately accessed in the downscaling step. A csv file named regions_months_global.csv is used to run 216 (18 sub-regions $\times$ 12 months) parallel jobs. This is a predefined list of sub-regions and months. Job script will go over the list of region-month combinations and will submit 216 parallel jobs.

3.1.1 Outputs

- If M is the number of coarse grid cells in the region (including the buffer region) T is the total number of days in the current and the downscaling period and the month is January, GCM data will be saved in a csv file named GCM_Jan.csv of the form of $M \times (T + 2)$ matrix. First two additional columns will provide lon/lat information.
- If N is the number of fine pixel locations in the region and T_o is the current observational period, ERA data matrix will be saved in a era_Current.Jan.csv file of the form of $N \times (T_o + 2)$ matrix.First two additional columns will provide lon/lat information.
- Two additional files named fine.pixels.exact.csv which includes lat/lon information of fine pixels included only in the exact region and fine.pixels.exact_index.csv which lists indexes of those exact pixels will be saved to identify fine pixels included in the exact region (excluding buffer) later in the downscaling step.
- Here we also performs standard downscaling and save it as Stand_SD.Jan.csv in the form of $N \times T$ matrix

3.1.2 Steps

- Change the paths as necessary in Preprocess.sh and Preprocess.R
- Specify the years to perform downscaling in Preprocess.R
- Command :
  qsub -v model="",scenario="",GCM_variable="",era_variable="",era_variable_spec=""
  Preprocess

3.2 BGL Downscaling

This step will produce downscaled projections and will save them as separate csv files. All the csv files saved at pre-prossessing step will be used here. First, code calculates temporal mean of the standard downscaled process and it will be saved in temporary memory to be later used in matlab executable. At the end, it will start to fit the model and perform downscaling. Each day will be saved as a separate .csv files within the respective sub-directory.

3.2.1 Outputs

- For each day, there will a csv files containing point estimates and uncertainty estimates.

3.2.2 Steps

- Change the paths as necessary in Downscaling.sh and Downscaling.R
- Specify the years to perform downscaling in Downscaling.R
- Command :
  qsub -v model="",scenario="",GCM_variable="",era_variable="",era_variable_spec=""
  Downscaling

3.3 Combine into netcdf file

Finally, all the date-specifc .csv file will be combined into .nc files.

3.3.1 Outputs

- Yearly netcdf files will be created including all the pixels arounf globe. Each file will include both point estimates and their uncertainty estimates.

3.3.2 Steps

- Change the paths as necessary in NC.file.sh and NC.file.R
- Specify the years to perform downscaling in NC.file.R
- Command :
  qsub -v model="",scenario="",GCM_variable="",era_variable="",era_variable_spec=""
  NC.file