

## **Wikiprint Book**

**Title:** Bias correction of seasonal forecasting data

**Subject:** TracMeteo - udg/ecomms/RPackage/biascorrection

**Version:** 12

**Date:** 07/06/2022 02:53:10 AM

## Table of Contents

Bias correction of seasonal forecasting data	3
Multi-variable bias correction	4

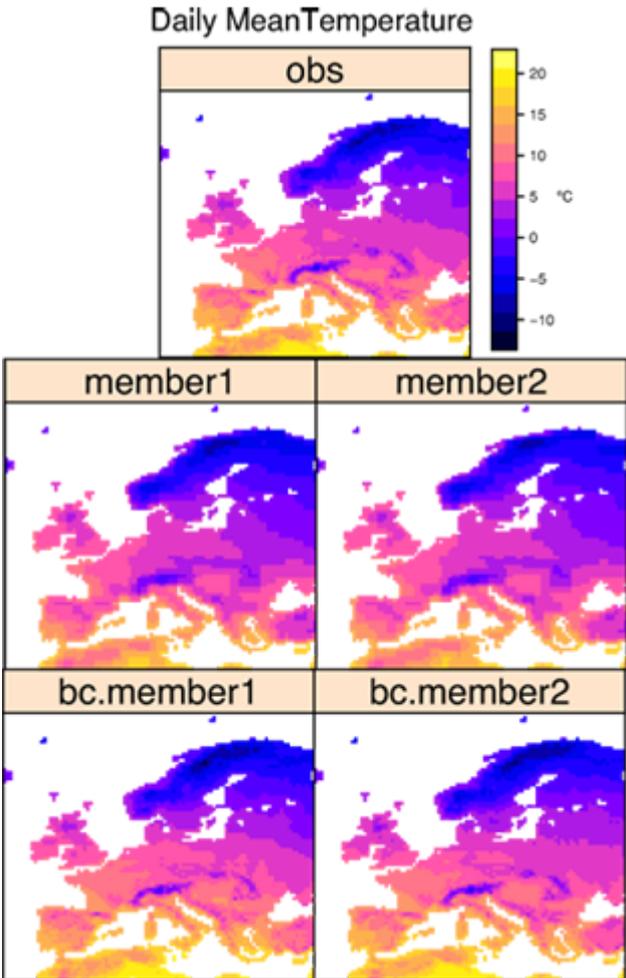
## Bias correction of seasonal forecasting data

**NOTE:** The following examples may take more than 30 min of computation. They have been conceived to provide an overview of the `loader.ECOMS` capabilities in combination with the companion tools for climate data analysis and post-processing. For a quicker overview, consider reducing the spatial/temporal domain of the initial request. Faster bias correction examples applied to seasonal forecast predictions are available in the [?downscaleR's wiki](#)

The seasonal forecasting data obtained from the `loader.ECOMS` package can be easily bias corrected (and downscaled) using the [?downscaleR](#) package (see a description of the [?bias correction functions](#)). This package has been developed in the framework of the SPECS and EUPORIAS projects for bias correction and downscaling of daily climate model outputs (with special focus in seasonal forecasting).

The following panels show an illustrative use of `ECOMS-UDG` and `downscaleR` to obtain the bias corrected series of mean temperature for the period DJFMAM (one-month lead time; i.e. with the initializations from November) over Europe. WFDEI is used as reference.

Note that, in order to facilitate the use of the resulting bias corrected data in different impact applications, the resulting bias corrected data can be easily exported to NetCDF format using the [?loader.2nc](#) extension.

R code	Output
<pre> library(loader.ECOMS) # Used for remote data access library(downscaleR) # Used for interpolation and bias correction library(loader.2nc) # Used to write NetCDF # login to UDG loginUDG(username = "username", password = "password")  # Loading data (two members) obs &lt;- loadECOMS(dataset = "WFDEI",   var = "tas",   season = c(12,1,2,3,4,5),   lonLim = c(-15,35),   latLim = c(32, 75),   years = c(2001:2010)) prd &lt;- loadECOMS(dataset = "System4_seasonal_15",   var = "tas", time = "DD",   season = c(12,1,2,3,4,5),   members = 1:2,   leadMonth = 1,   lonLim = c(-15,35),   latLim = c(32, 75),   years = c(2001:2010))  # Parallelization is used to speed-up interpolation: prd &lt;- interpGrid(prd, new.coordinates = getGrid(obs),   method = "bilinear",   parallel = TRUE)  # Bias correction and plotting prd.bc &lt;- biasCorrection(obs, prd, prd,   method = "eqm",   window = c(30,10))  plotMeanGrid(obs) plotMeanGrid(prd, multi.member = FALSE) plotMeanGrid(prd.bc, multi.member = FALSE)  # Exporting to netcdf4 fileName &lt;- "tas_qqmap_System4_WFDEI_2001_2010.nc4" grid2nc(data = prd.bc,   NetCDFOutFile = fileName,   missval = 1e20,   prec = "float") </pre>	 <p>The output displays four maps of Europe's daily mean temperature. The top map, labeled 'obs', shows the observed data with a color scale from -10 to 20 °C. Below it are four panels: 'member1', 'member2', 'bc.member1', and 'bc.member2'. The 'bc' panels show the bias-corrected results, which appear more similar to the 'obs' panel than the original model members.</p>

The `netcdf4` file resulting from this example can be downloaded here:

[http://www.meteo.unican.es/work/datasets/tas\\_qqmap\\_System4\\_WFDEI\\_2001\\_2010.nc4](http://www.meteo.unican.es/work/datasets/tas_qqmap_System4_WFDEI_2001_2010.nc4)

In order to take into account the model drift (the change of the model bias as a function of the lead time), the bias correction methods are applied considering the lead month of the predictions as an extra dimension. This is implemented in the `biasCorrection` function by considering a moving

time window and a time step for (one month in the above example and a time step of 10 days: `window = c(30,10)`, in days). The default recommended value for `window` is one month, although some tests are being conducted in order to determine the optimum window to correct the available seasonal forecasts (further information will follow).

## Multi-variable bias correction

Since a number of variables are typically required in impact applications (in particular in EUPORIAS WP23 and WP31; [available variables](#)), we recommend two alternative bias correction methodologies for these tasks: a) the ISI-MIP methodology, b) qqmap bias correction.

In order to facilitate this task (multi-variable bias correction), an script has been prepared to correct the following variables:

`ps,wss,huss,tas,tasmax,tasmin,tp,rsds,rlds` (the above codes correspond to the standard names used in `loadECOMS` and `downscaleR` packages). The example below applies the qqmap technique (considering the WFDEI observations) to the 15 members of the System4 dataset, for the six-month series (DJFMAM) corresponding to the November initialization for the period 2001-2010 in an European domain. The resulting bias corrected series are stored in a separate netcdf file.

```
# Seasonal forecast parameters
dataset <- "System4_seasonal_15"
season <- c(12,1:5)
leadMonth <- 1
members <- 1:15
lonLim <- c(-15,35)
latLim <- c(32, 75)
years <- c(2001:2010)
time <- "DD"

# Bias correction parameters
interpolationMethod <- "nearest" # Both observation and forecast should be define on the same grid. Options: "nearest" and
method <- "eqm" # Empirical quantile mapping
pr.threshold <- 1 # The minimum value that is considered as a non-zero precipitation.
window <- c(30,10) # Integer vector specifying the time window width and the time step used to calibrate. The window is ce
```

```
# Script to bias correct and store (as netcdf file) the data
variables <- c("ps","wss","huss","tas","tasmax","tasmin","tp","rsds","rlds")
for (v in 1:length(variables)){
  obs <- loadECOMS(dataset = "WFDEI",
                  var = variables[v],
                  lonLim = lonLim,
                  latLim = latLim,
                  season = season,
                  years = years[1])

  prd <- loadECOMS(dataset = dataset,
                  var = variables[v],
                  time = time,
                  members = members,
                  lonLim = lonLim,
                  latLim = latLim,
                  season = season,
                  years = years,
                  leadMonth = leadMonth)

  prd <- interpGrid(prd,
                  new.coordinates = getGrid(obs),
                  method = interpolationMethod)

  prd <- if ("tp" %in% obs$Variable$varName) {
    biasCorrection(obs,
                  prd,
                  prd,
                  pr.threshold = pr.threshold,
```

```
                method = method,
                window = window)
} else {
    biasCorrection(obs,
                  prd,
                  prd,
                  method = method,
                  window = window)
}
# Exporting to netcdf4
fileName <- paste(var[v], "System4_WFDEI.nc4", sep = "_")
grid2nc(prd, NetCDFOutFile = fileName,
        missval = 1e20,
        prec = "float")
}
```