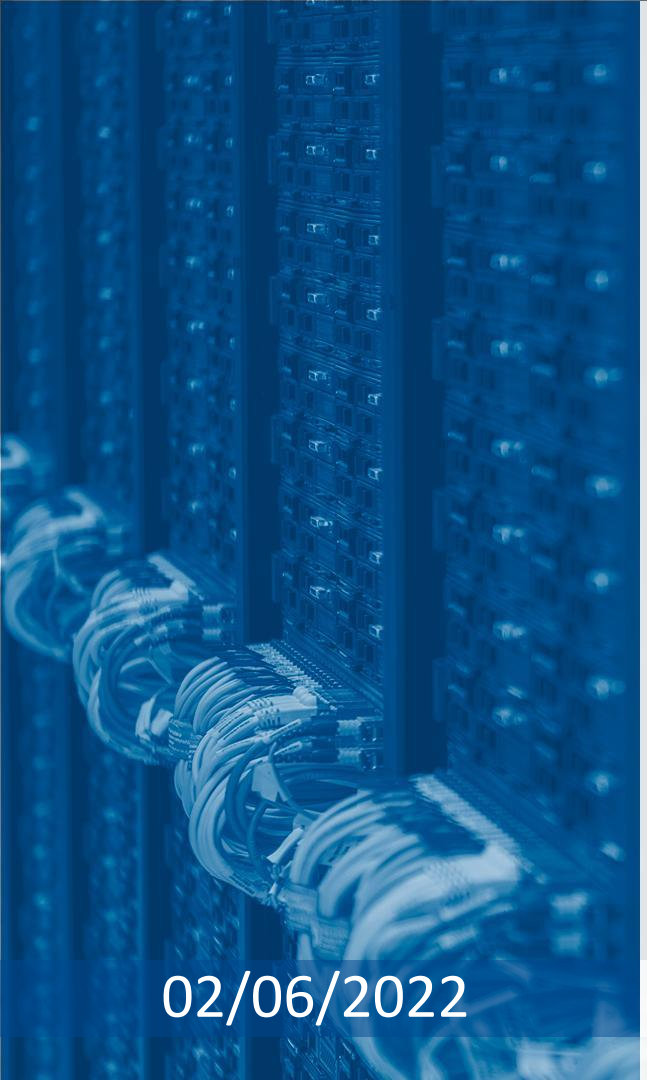


Barcelona Supercomputing Center
Centro Nacional de Supercomputación

R user meeting

An-Chi Ho

contributor: Jaume Ramon

02/06/2022

Agenda

1. Ice-breaker
2. News
   - s2dv
   - ClimProjDiags
   - General R
3. User presentation: CSDownscale package [Jaume]
4. Q&A
   -

Ice-breaker

No ice-breaker today :(

IT periodic talk yersterday (1st June)

https://earth.bsc.es/wiki/lib/exe/fetch.php?media=computing:it_periodic_talks_hpc_-_01-06-2022.pdf

s2dv

MeanDims() efficiency improvement

s2dv::MeanDims is improved to have better efficiency.

Status: Master branch

!! Note that apply() could be very slow when operation is over large dimensions !!

```
exp <- array(rnorm(10000), dim = c(dat = 2, memb = 4, sdate = 120, lat = 90, lon = 60))
MeanDims(exp, c('dat', 'memb'))
```

new s2dv::MeanDims

Code	File	Memory (MB)		Time (ms)
▼ MeanDims2	<expr>	0	0	80
aperm	MeanDims2.R	0	0	80

old s2dv::MeanDims

Code	File	Memory (MB)		Time (ms)
▼ MeanDims1	<expr>	-133.4	155.9	6180
▼ apply	MeanDims1.R	-133.4	155.9	6180
▶ FUN		-86.0	78.8	3770
array		-24.3	45.1	1840
aperm		0	0	80

MeanDims() efficiency improvement

The functions that use MeanDims() inside are more efficient now.

E.g., ACC()

```
exp <- array(rnorm(100000), dim = c(dataset = 1, lat = 30, lon = 20, member = 10, sdate
= 120))
obs <- array(rnorm(100000), dim = c(dataset = 1, lat = 30, lon = 20, member = 1, sdate =
120))
```

new s2dv::ACC

Code	File	Memory (MB)		Time (ms)
▼ ACC	<expr>	0	38.2	480
▼ compiler:::tryCmpfun	<expr>	0	3.2	200
▶ tryCatch		0	3.2	200
▶ Apply		0	32.4	200
▶ MeanDims		0	2.6	80

old s2dv::ACC

Code	File	Memory (MB)		Time (ms)
▼ ACC	<expr>	-49.4	69.1	1370
▶ MeanDims		-49.4	54.2	1170
▶ Apply		0	14.9	200

PlotEquiMap new parameters

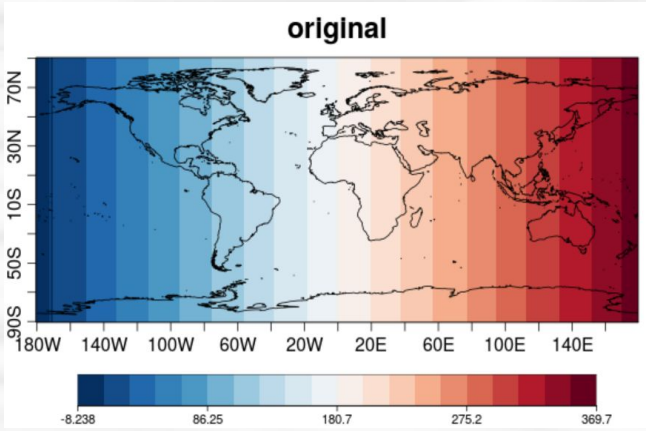
New parameters **`xlabels`**, **`ylabels`**, **`xlonshft`**, **`ylatshft`** to improve the tick labels, especially when longitudes and latitudes are shifted.

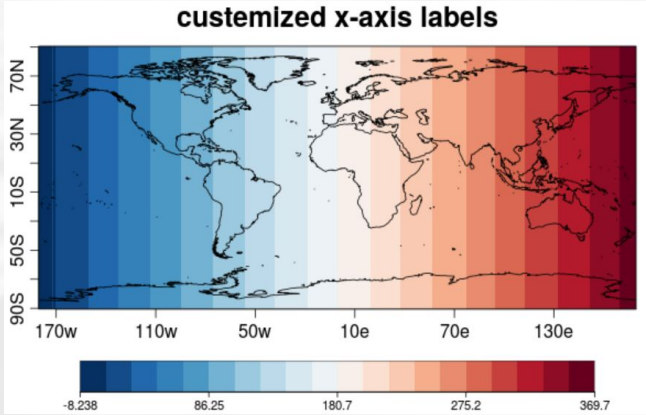
The default sets the first x tick label at 0 (left margin), and **`xlonshft`** shifts the ticks by the degrees specified, e.g. `xlonshft = 10` will shift the ticks by 10 degrees.

With **'xlabels'**, you can customize the value of longitude labels. The number of values depend on the parameter 'intxlon', which is the interval between longitudes to label. The main point of this parameter is to avoid the wrong labels (see plot a) when shifting the lons.

Status:
https://earth.bsc.es/gitlab/es/s2dv/-/tree/develop_PlotEquiMap

```
xlonshft=10,intxlon=60,xlabels=c(paste0(seq(170,  50, by = -60), 'w'), paste0(seq(10, 190, by = 60), 'e'))
```





PlotEquiMap shift map

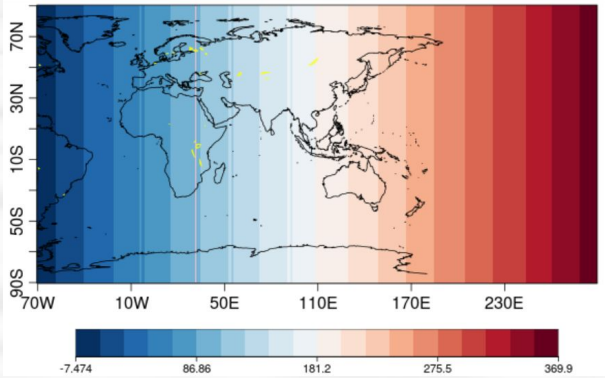
The current PlotEquiMap() can only plot map with either [0, 360] or [-180, 180]. The new development allows flexible longitude range.

Status:
https://earth.bsc.es/gitlab/es/s2dv/-/tree/develop_PlotEquiMap

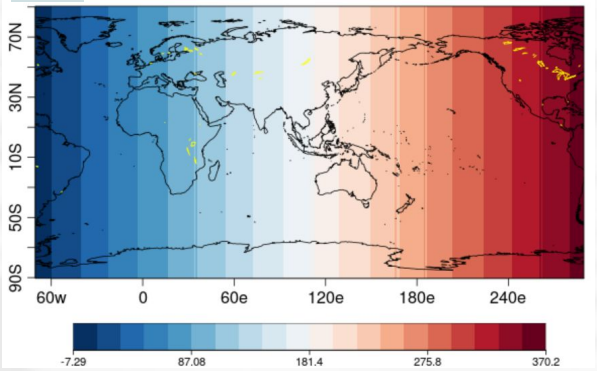
****Since the code of plotting map is changed quite a lot, it is recommended testing it with your script before it's officially included in the package.**

```
data <- array(1:360, dim = c(lon = 360, lat = 181))
lon <- -180:179 + 110   #shift 110deg
lat <- -90:90
PlotEquiMap(data,lon,lat,filled.continents=F,
country.borders=F,filled.ocean=F,lake_color='yellow',
xlonshft=10,intxlon=60,
xlabels=c('60w','0',paste0(seq(60,300,by=60),'e')))
```

old



new



Load(): Bugfix for R_4.1.2

Load() returns error in R_4.1.2:

Error in prod(dim_exp) : invalid 'type' (list) of argument

Status: Master branch

- s2dverification won't be fixed. So it is only available for R < 4.0.0.
- CSTools::CS_Load uses s2dverification::Load for now. Before it is changed to s2dv, it can't perform well in R_4.1.2 either.

ClimProjDiags

WeightedMean() bugfix

dlon calculation correction

https://earth.bsc.es/gitlab/es/ClimProjDiags/-/issues/7

Status: Master branch

Use it now by

```
source("https://earth.bsc.es/gitlab/es/ClimProjDiags/-/raw/master/R/WeightedMean.R")
```

General R

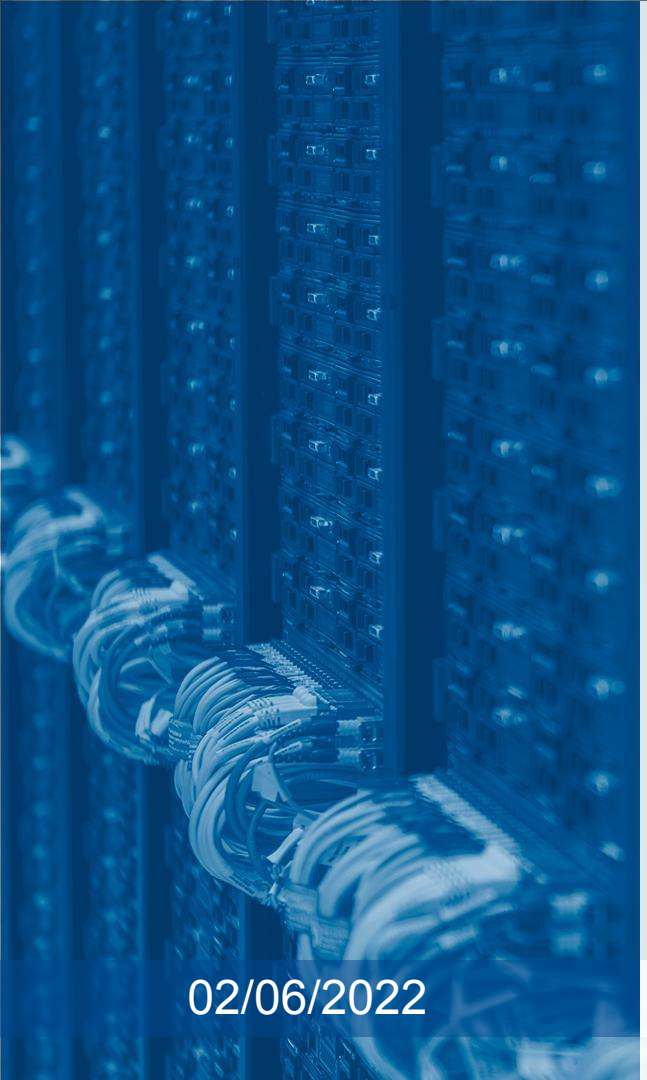
General R

- Start using **module R/4.1.2** on Workstation and Nord3v2, and report problems if found.
- RStudio on WS: https://earth.bsc.es/wiki/doku.php?id=computing:workstations#using_rstudio-server_in_ws

RStudio should have the same environment (same modules loaded, same directories, etc.) as your workstation. If you find them different, restart RStudio by:

- Press command/ctrl + shift + F10 (with fn)
- Type `.rs.restartR()`

CSDownscale



Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Statistical downscaling with CSDownscale

Jaume Ramon

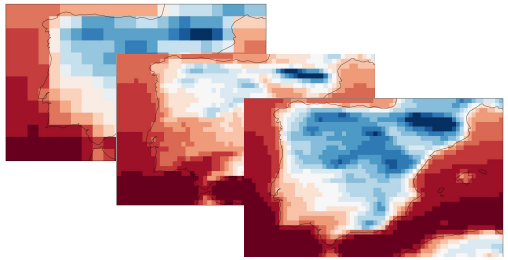
With inputs from Llorenç, Carlos D., Lluís, Núria, Raül, An-Chi, Alba…

02/06/2022

R user meeting

CSDownscale

- Will be an **open source** R package.
- The user should be able to easily **compare** the performance of different **downscaling methods**.



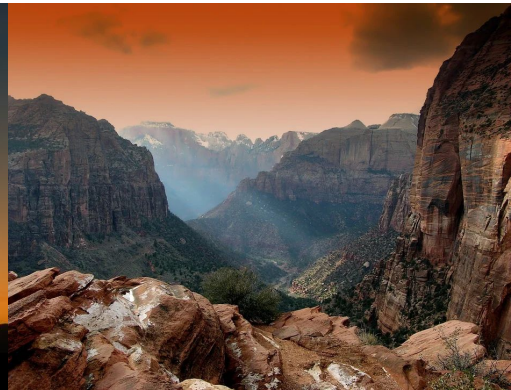
Downscaling methods

1) Interpolation
2) Interpolation plus bias adjustment
3) Interpolation plus linear regression
4) Large-scale predictors and local climate variables
5) Stencil
6) Analogs
7) Logistic regression

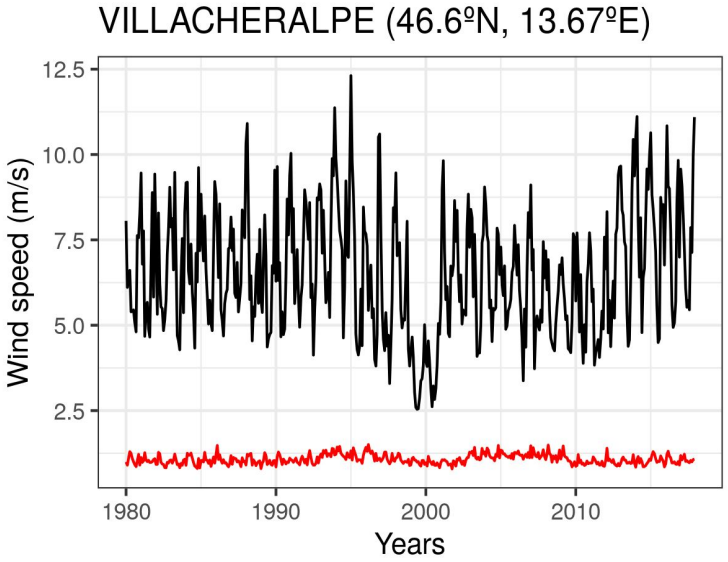
Why downscaling?

- Downscaling is any procedure to infer **high-resolution** information from **low-resolution** variables.
- Climate predictions are delivered on **grids** of thousands of square kilometres, which is little useful for **local applications** (e.g. predictions at a wind farm, river basin or mountain valley).
- Go from coarse to fine grids, or even to a **point scale**.





Why downscaling?



In situ observations

ERA5

Functions in CSDownscale

```
Interpolation <- function(exp, points = NULL, method_remap = NULL, target_grid = NULL, lat_dim = "lat",
                          lon_dim = "lon", remap_region = NULL, method_point_interp = NULL)
```

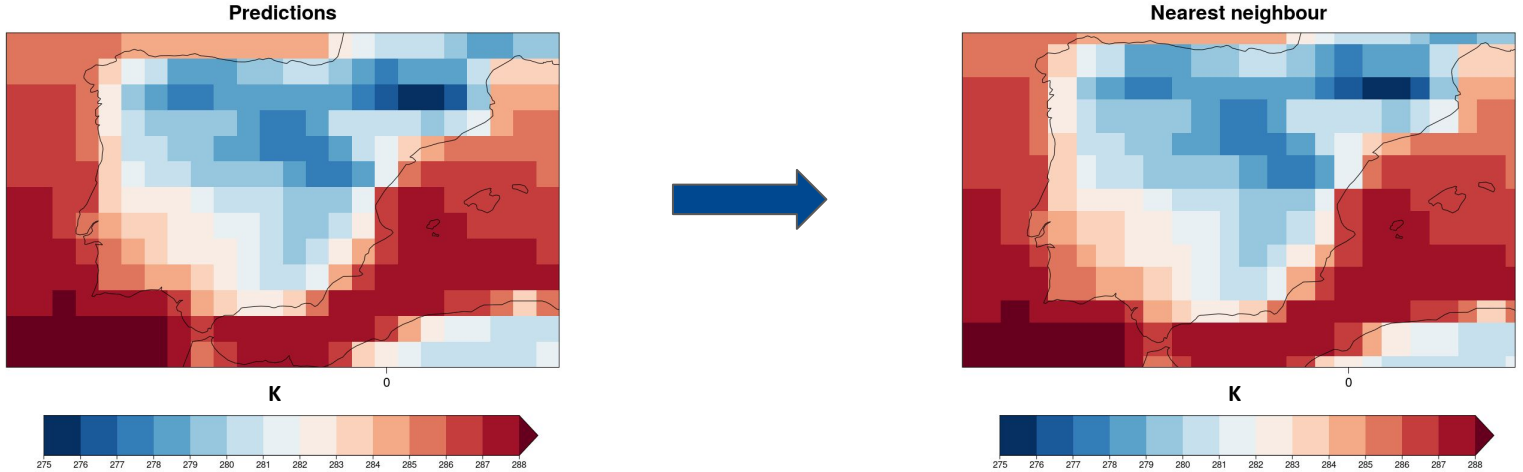
```
Intbc <- function(exp, obs, target_grid, int_method, bc_method, points = NULL, method_point_interp = NULL,
                  lat_dim = "lat", lon_dim = "lon", sdate_dim = "sdate", member_dim = "member", ncores = 1)
```

```
Intlr <- function(exp, obs, target_grid, int_method, lr_method, predictors = NULL, lat_dim = "lat",
                  lon_dim = "lon", sdate_dim = "sdate", time_dim = "time",
                  large_scale_predictor_dimname = 'vars', loocv = FALSE, ncores = 1)
```

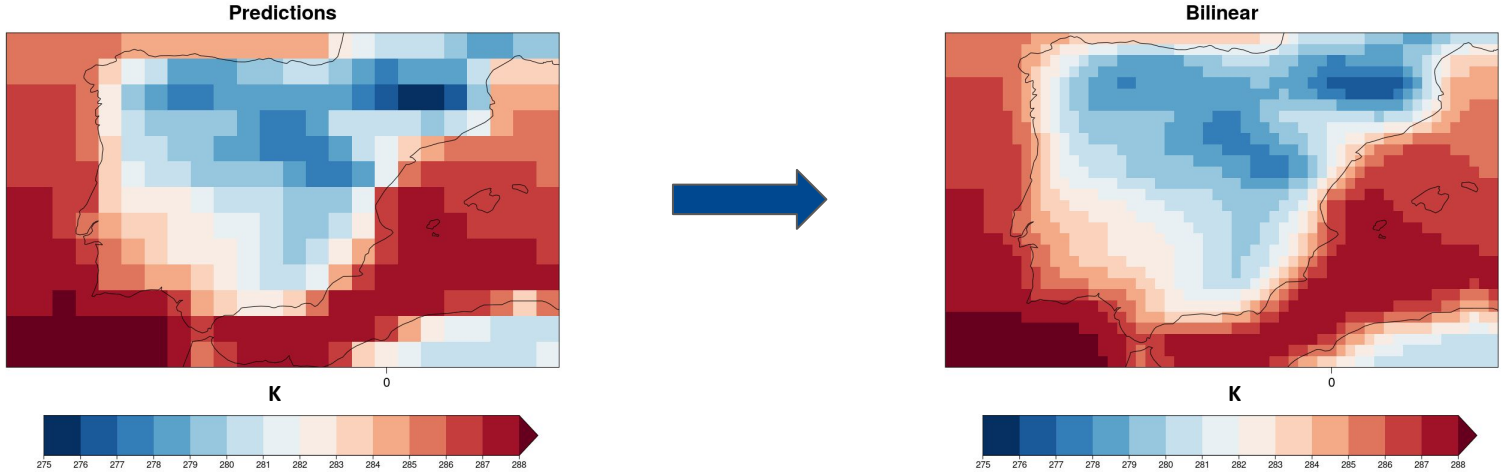
```
Analogs <- function(exp, obs, nanalogs = 3, fun_analog = "mean", lat_dim = "lat", lon_dim = "lon",
                  sdate_dim = "sdate", time_dim = "time", grid_exp = NULL, ncores = 1)
```

Interpolation(..., method = “nearest_neighbour”)

- Regrid from coarse to fine grid.

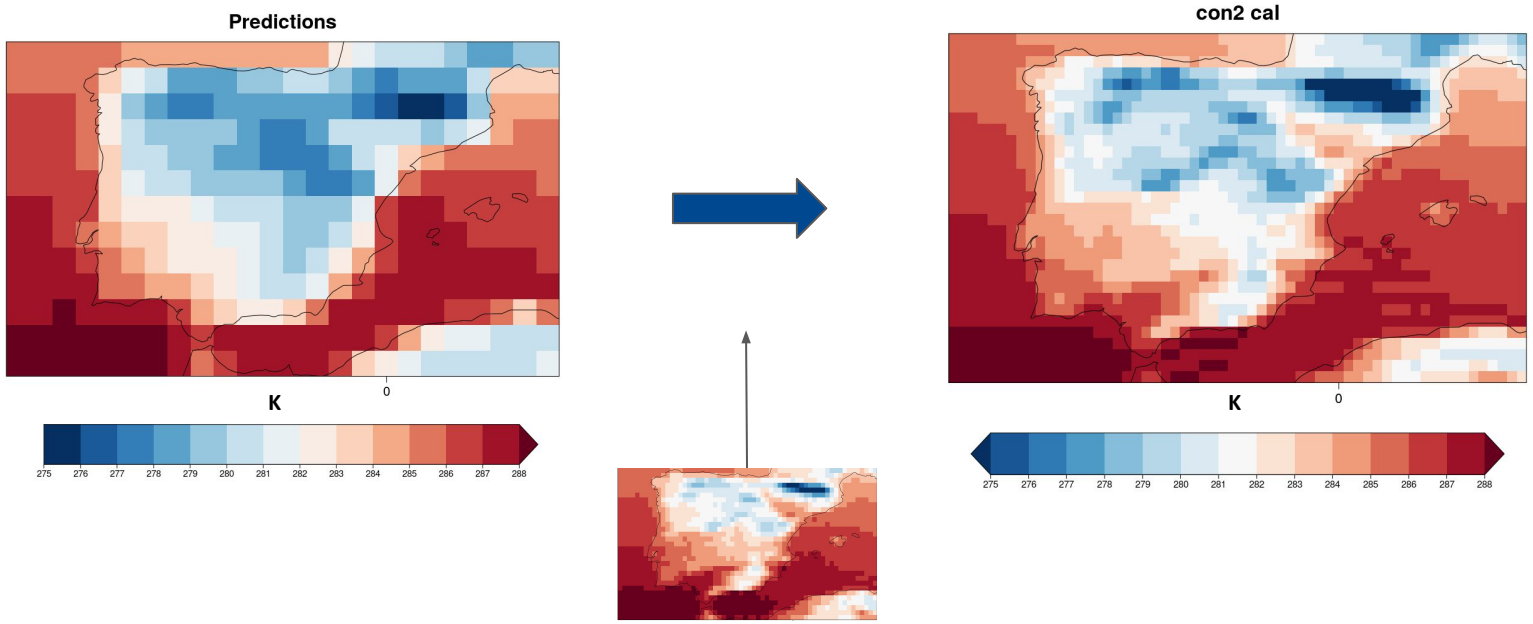


Interpolation(..., method = “bilinear”)



Intbc(..., bc_method = “calibration”**)**

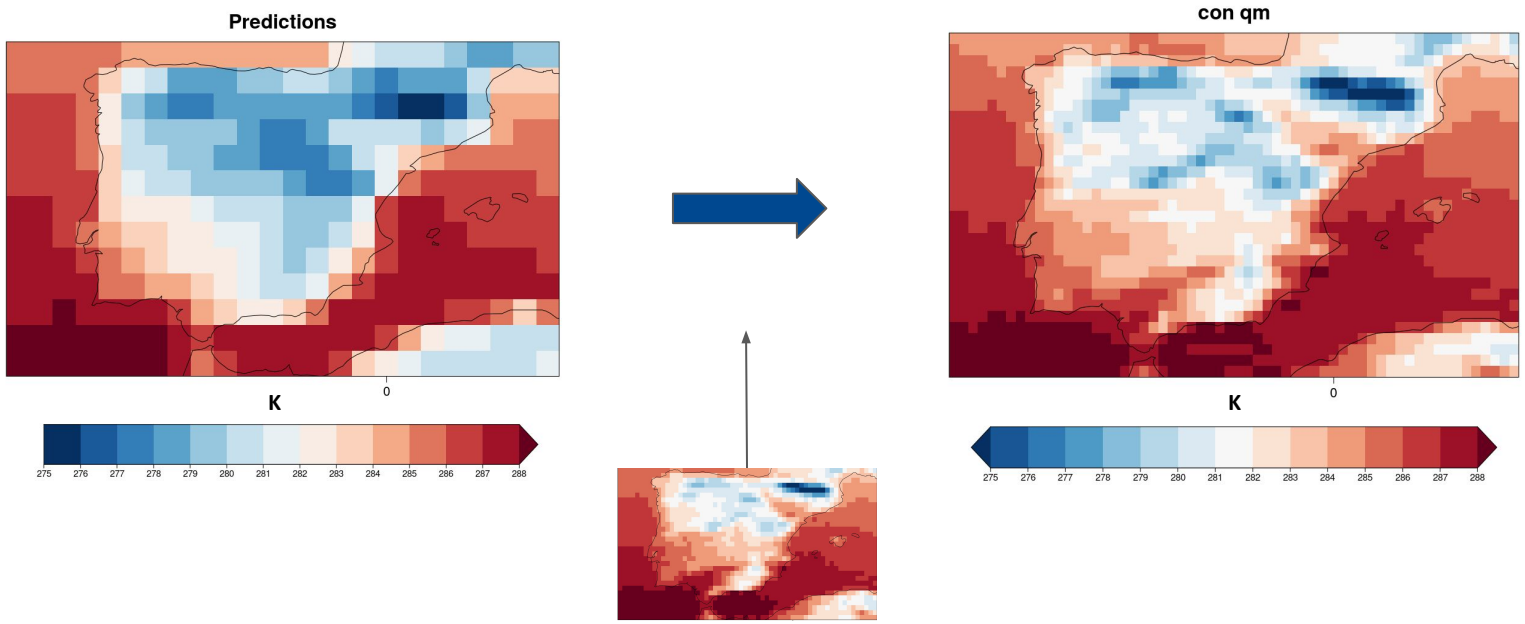
- First, interpolate from coarse to fine grid. Then, do a bias adjustment.



High-res observations

Intbc(..., bc_method = “quantile_mapping”**)**

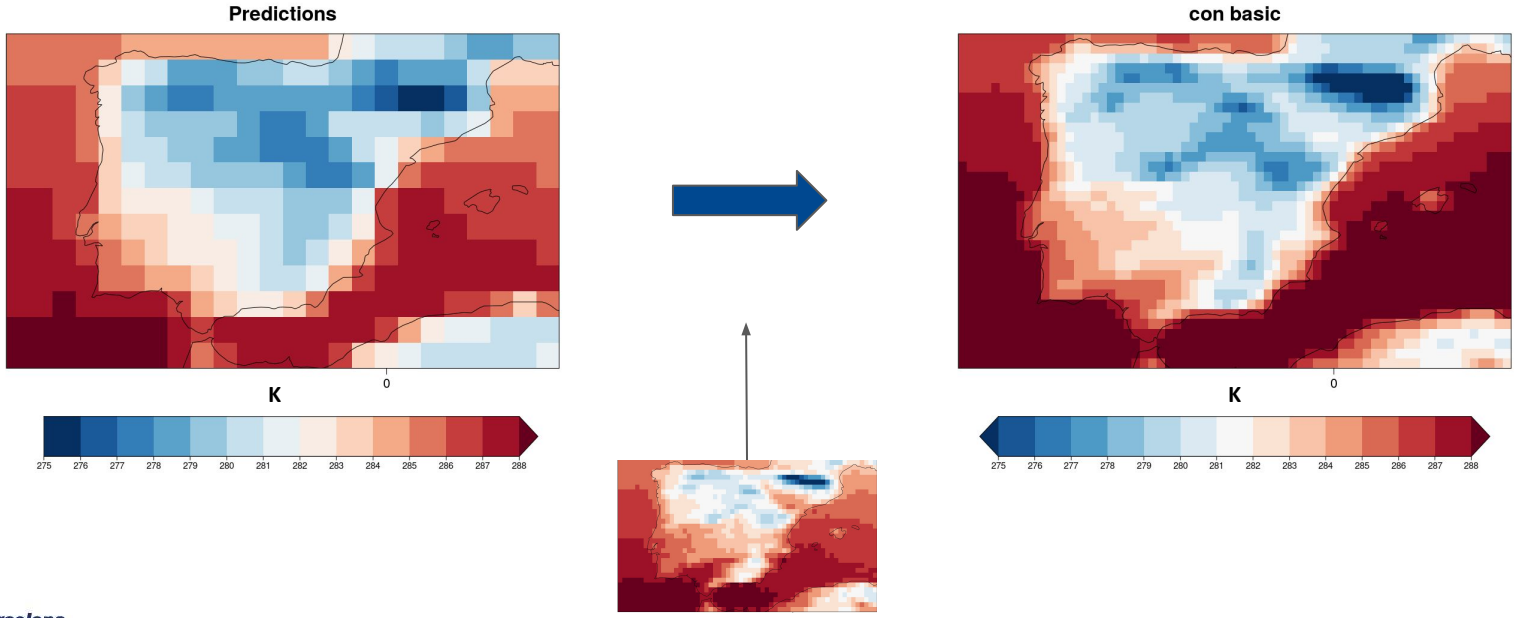
- First, interpolate from coarse to fine grid. Then, do a bias adjustment.



High-res observations

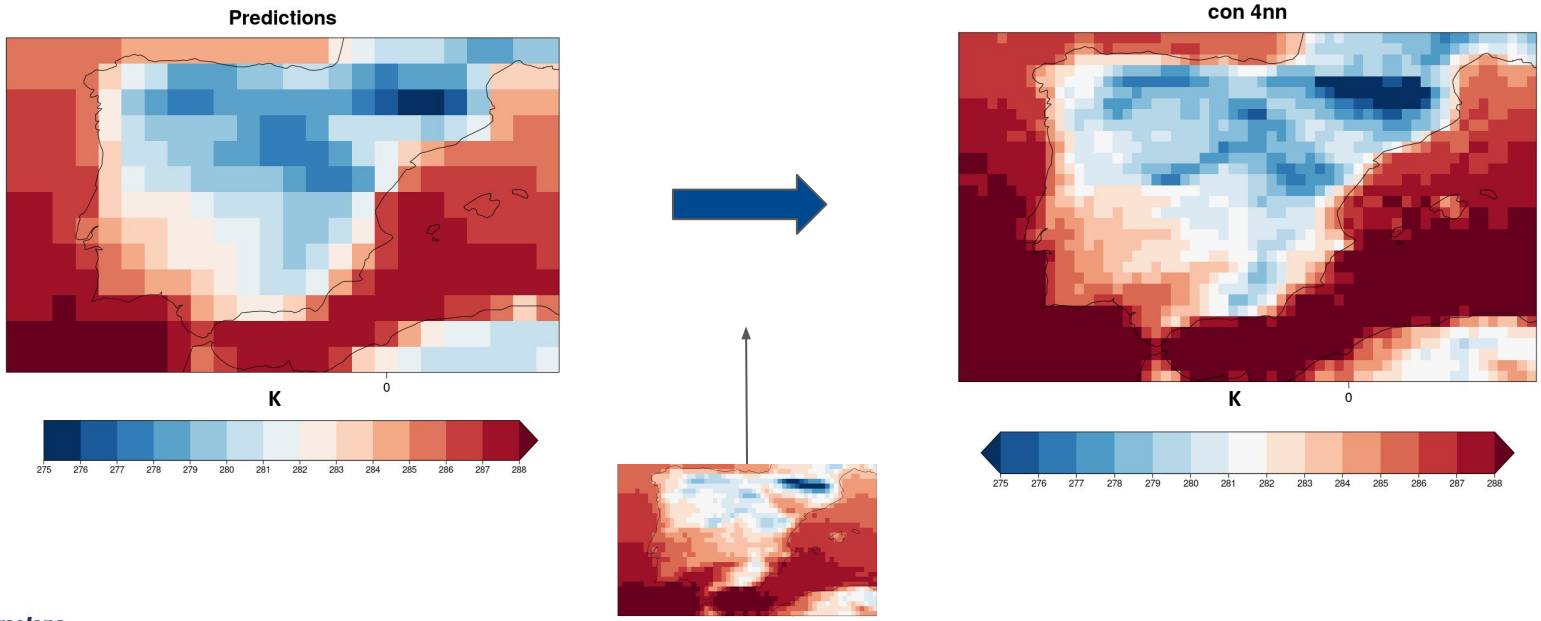
IntIr(..., lr_method = “basic”**)**

- First, interpolate from coarse to fine grid. Then, adjust the models with a point-wise linear regression with the high-res observations as predictands.



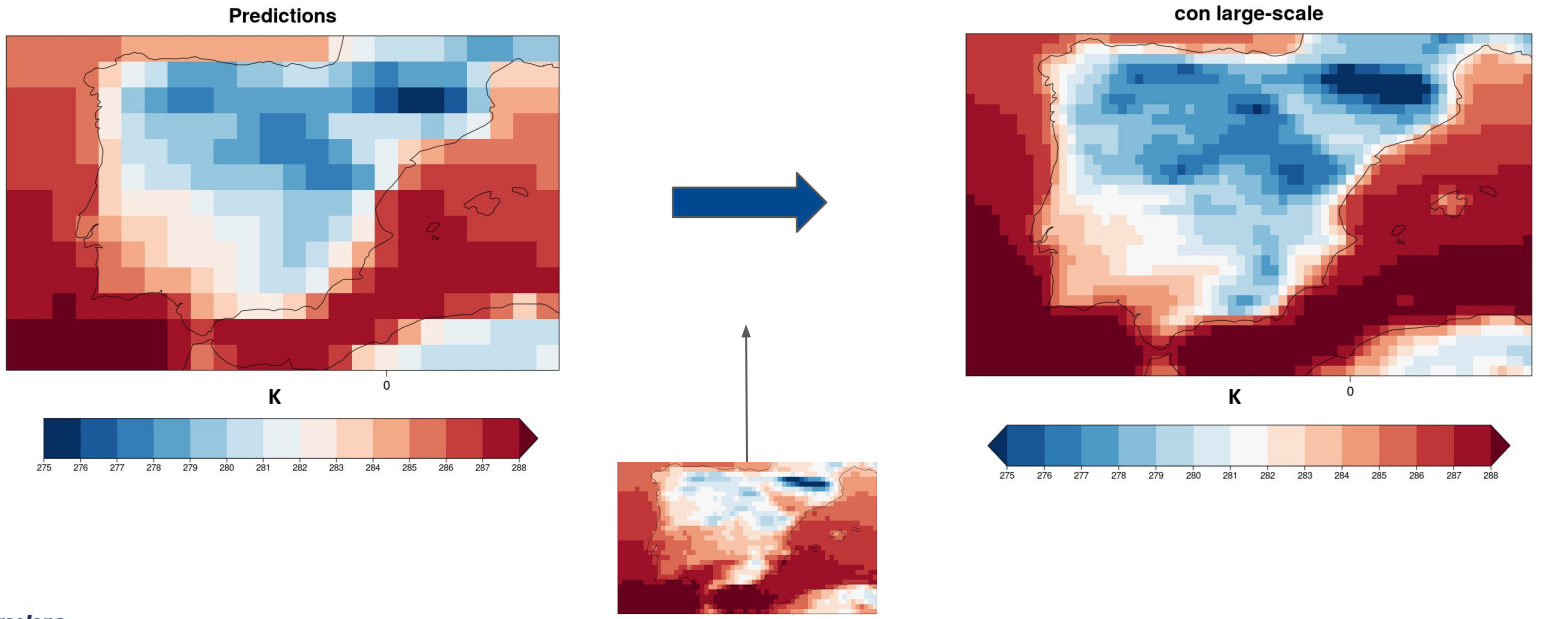
IntIr(..., lr_method = “4nn”**)**

- First, interpolate from coarse to fine grid. Then, adjust the models with a point-wise multi-linear regression with the four NN as predictors.



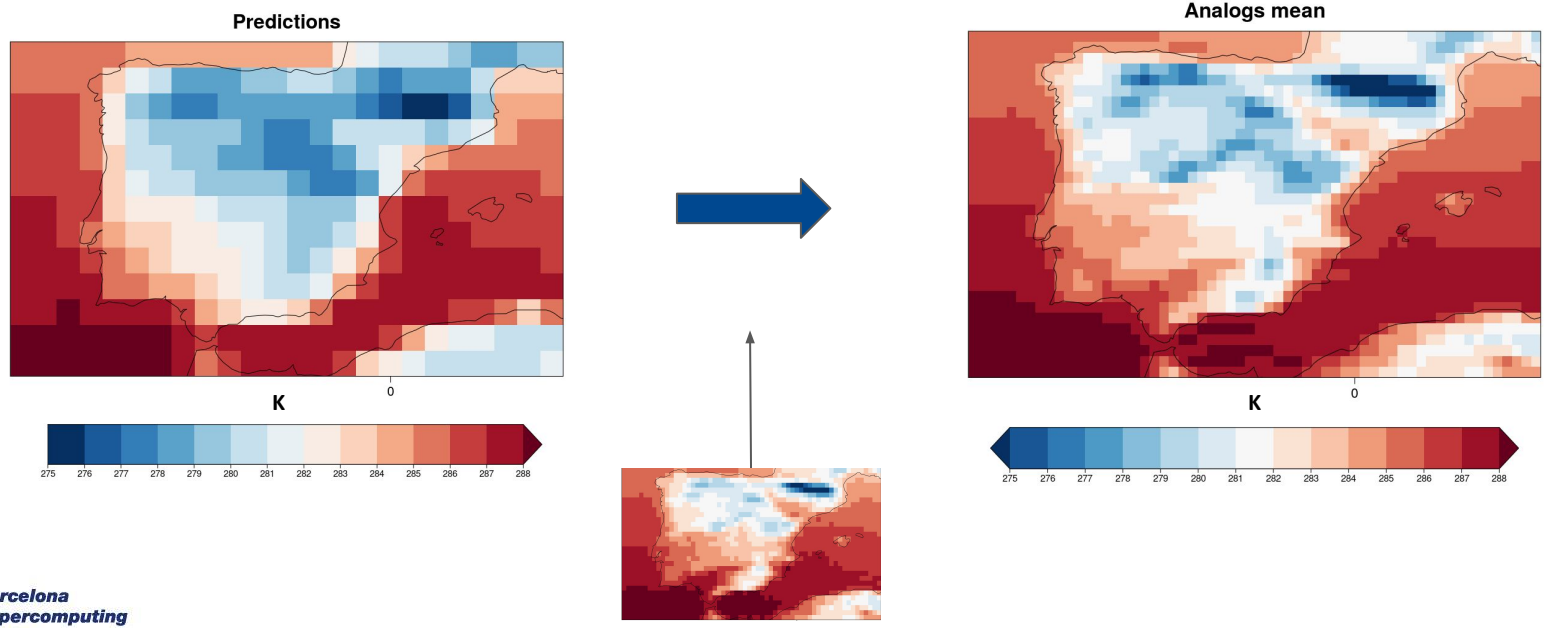
Intlr(..., lr_method = “large-scale”**)**

- First, interpolate from coarse to fine grid. Then, adjust the models with a point-wise multi-linear regression with large-scale predictors (e.g. teleconnection indices).



Analogs(..., nanalogs = 3, fun_analogs = 'mean'**)**

- Interpolate high-res observations to model grid. Then, compute the Euclidean distance between the model and observations' fields. Select the best analogs in the high-res observational fields.



Q & A

Thanks for joining

Next meeting: 7th July 2022 (11 am)