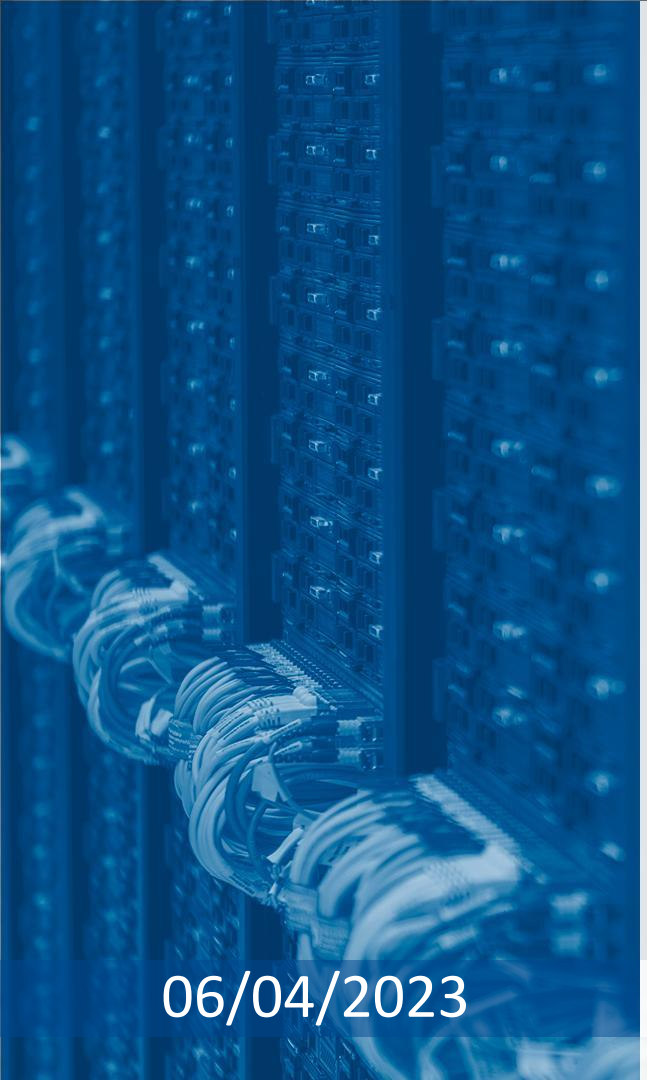


Barcelona Supercomputing Center
Centro Nacional de Supercomputación

R user meeting

An-Chi Ho, Eva Rifà, Victòria Agudetse

contributor: Nadia Milders

06/04/2023

Agenda

1. Ice-breaker: Memory profiling
2. News
   - General R
   - ClimProjDiags
   - s2dv
   - startR
   - multiApply
   - CSTools
   - CSIndicators
   - Verification Suite
3. User presentation: New function “PlotRobinson” [Nadia]
4. Q&A

Ice-breaker


BSC
Barcelona
Supercomputing
Center
Centro Nacional de Supercomputación

Packages and functions for memory profiling

- profvis package

- Memory: Memory allocated or deallocated (for negative numbers) for a given call stack. This is represented in megabytes and **aggregated** over all the call stacks over the code in the given row.
- Time: Time spent in milliseconds. This field is also **aggregated** over all the call stacks executed over the code in the given row
- More details in the previous meeting slides (page 12-14) made by Núria
- Tip: Sourcing the function file (instead of calling function from package) can show the profiling of each line.

Packages and functions for memory profiling

- peakRAM package

  Small package with one function to tell you what's the peak RAM in a given chunk of code.

- memuse package
  - Nice user guide
  - Useful functions: Sys.filesize, Sys.meminfo, Sys,procmem, memuse

```
> memuse::Sys.filesize("/esarchive/exp/ecmwf/system5c3s/monthly_mean/tas_f6h/tas_19810101.nc")
26.647 MiB
> memuse::Sys.meminfo()
Totalram:  15.383 GiB   # Nord3-standard node: 32Gb; medmem node: 64Gb
Freeram:    6.946 GiB
> memuse::Sys.procmem()
Size:  180.852 MiB
Peak:  180.852 MiB
> memuse(res, unit = 'best')
8.922 KiB
```

Some comparisons

(1) RAM used

- **memuse::Sys.procmem** shows the amount of ram used by the current R process
- **pryr::mem_used** shows how much memory is currently used by R. Sum-up of gc()

```
> pryr::mem_used()
31.2 MB
> memuse::Sys.procmem()
Size:  66.734 MiB
Peak:  66.734 MiB
```

(2) peak RAM

- **peakRAM::peakRAM** monitors the total and peak RAM used by any number of R expressions or functions
- **memuse::Sys.procmem** shows the amount of ram used by the current R process

```
> peakRAM::peakRAM({d <- func(10000)})

     Function_Call Elapsed_Time_sec Total_RAM_Used_MiB Peak_RAM_Used_MiB
1 {d <- func(10000)}           0.001                 0.1                0.2
```

Some comparisons

(3) Data size

- **utils::object.size**
- **pryr::object_size** is more accurate than object.size
- **memuse::memuse**

```
> object.size(data)
1600784 bytes
> format(object.size(data), unit = 'auto')
[1] "1.5 Mb"
> pryr::object_size(data)
1,600,784 B
> pryr::compare_size(data)
   base    pryr
1600784 1600784
> memuse::memuse(data)
1.527 MiB
```

Why do you need profiling?

Spend some time on profiling = save more time in the long term!

→ Check your script to find the efficiency bottleneck memory- or time-wise. If it happens in some functions, report in the corresponding GitLab.

→ Your tests would be more practical and meaningful than what we do.

→ Remember that multiApply could be heavy for light operation; try to use more cores and larger data to see if the performance makes sense.

General R

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

R community meeting

- **R community meetup at BSC**:
  - Date: April 20th at 18.30 h
  - Location: BSC Repsol Building Auditorium
- **Schedule:**
  - 18:30 h - Collaborative R tools for Climate Forecast Analysis by HPC
    (An-Chi and Eva, Computational Earth Sciences)
  - 19:00 h - AI in R: PredIG an explainable XGBoost predictor for cancer immunotherapy
    (Roc Farriol, Electronic and Atomic Protein Modelling)

@RLadiesBCN
@BarcelonaR_user
https://www.meetup.com/rladies-barcelona/

ClimProjDiags

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

New release 0.3.1

NEWS:

- Subset(): Prioritize the dimension names from names(dim(x)) rather than attribute ‘dimensions’; If the input data doesn’t have dimension names, the output doesn’t have either.

```
x <- array(rnorm(100), dim = c(a = 10, b = 2, c = 5))
attributes(x)$dimensions <- c(‘b’, ‘a’, ‘c’)
> str(x)
 num [1:10, 1:2, 1:5] 1.881 -0.666 -0.3 0.883 1.019 ...
 - attr(*, "dimensions")= chr [1:3] "b" "a" "c"
```

Now, Subset() uses dimension names c(“a”, “b”, “c”) instead of c(“b”, “a”, “c”).

Might be a problem on Load() outputs since the attribute “dimensions” is created by Load().

s2dv

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

New version 1.4.0

Include new functions, bugfixes, new features developed during the past months. Check news:
https://cran.r-project.org/web/packages/s2dv/news/news.html

NAO() parameter “ftime_avg”

NAO() parameter "ftime_avg" can be NULL so no average is calculated.

“**ftime_avg**”: A numeric vector of the forecast time steps to average across the target period. **If average is not needed, set NULL**. The default value is 2:4, i.e., from 2nd to 4th forecast time steps.

```
> res <- s2dv::NAO(exp = exp, obs = obs, lat = lat, lon = lon)
> dim(res$exp)
 sdate member
     3      2
> res <- s2dv::NAO(exp = exp, obs = obs, lat = lat, lon = lon, ftime_avg = NULL)
> dim(res$exp)
 sdate member  ftime
     3      2      5
```

status: in v1.4.0

Efficiency improvement of ProjectField() and RPSS()

Avoid using `apply(data, …, mean/sum)` since it is heavy. Use `colMeans/rowMeans/colSums/rowSums` instead.

The improvement in ProjectField() also benefits NAO() and EOF().

status: ProjectField() in v1.4.0; RPSS() in master

New function GetProbs()

Compute probabilistic forecasts or the corresponding observations.

Used in RPS, RPSS, ROCSS.

Check function: https://earth.bsc.es/gitlab/es/s2dv/-/blob/master/R/GetProbs.R

status: in master

Unify the functions regarding significance test

As discussed several months ago, we planned to make the inputs and outputs regarding significance test in all s2dv functions (if applicable) consistent.

The plan:

- Parameter “alpha” is numeric (0.05 by default), replacing “conf.lev = 0.95”
- Significant test outputs are “p.val”, “conf.lower”, “conf.upper”, “sign”
- Flag parameters “pval = TRUE”, “conf = TRUE”, “sign = FALSE” --> If TRUE, return the corresponding item.

status: in branch develop-alpha

issue: https://earth.bsc.es/gitlab/es/s2dv/-/issues/79#note_208573

All dat_dim default is changed to NULL

As discussed several months ago, if the function has parameter “dat_dim”, the default is NULL (except for a few functions that aim to calculate with multiple datasets)

status: in branch develop-alpha

issue: https://earth.bsc.es/gitlab/es/s2dv/-/issues/78

startR

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

New version 2.2.2

NEWS: https://cran.r-project.org/web/packages/startR/news/news.html

- Start(): Bugfix when the input parameters are assigned by a variable with NULL value and retrieve = FALSE

```
it_is_null <- NULL
data <- Start(dat = …,
              transform = it_is_null,
              transform_params = NULL,
               …,
              retrieve = FALSE)
Error in start_call[[i]] : subscript out of bounds
```

multiApply

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

New version 2.1.4

More safety checks to ensure the output is correct.

If you use the function “correctly”, you have no problems.

Mind the warnings

When using Apply() or the functions that use Apply(), if you encounter the warning like:

```
In arrays_of_results[[component]][(1:prod(component_dims)) +  ... :
  number of items to replace is not a multiple of replacement length
```

or other warnings not intendedly produced by Apply(), it probably has problems.

→ Check the function used in Apply(). *Does the output has the same dimensions all the time?*

```
data <- array(1:12, dim = c(time = 4, member = 3))
res <- Apply(data, fun = mean, target_dims = 'time')
```

What does it mean? → mean()'s input is a 1-dim array [time = 4] and it is run 3 times (margin dim [member = 3]). So, the outputs of the 3 times should have the same dimensions.

CSTools

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

New release 5.0.0

- On CRAN: https://cran.r-project.org/web/packages/CSTools/
- Installed in workstation & Nord3v2, R/4.1.2-xxx.
- Check the NEWS:
  - New `s2dv_cube` object development
  - New plotting function **PlotWeeklyClim**
  - New function **CST_Subset**
  - New function **CST_InsertDim**
  - Allow `memb_dim` to be NULL in **QuantileMapping**
  - Add `dat_dim` parameter in **BiasCorrection** and **Calibration**
  - Correct vignettes: **Analogs**, **MultiModelSkill** and **MultivarRMSE**

CST_Subset

- A wrapper of **ClimProjDiags::Subset** for `s2dv_cube` objects
- Same parameters as in **Subset**, plus **var_dim** (variable dimension name) and **dat_dim** (dataset dimension).
- **Specifications:**

  $data: A simple application of ClimProjDiags::Subset
  $dims: The dimensions in $data are updated
  $coords: Coordinates values and dimensions subset and removed if they are dropped
  $attrs:
  - **$Dates:** subset along time dimensions.
  - **$source_files**, **$Datasets** and **$Variables** are Subset along the corresponding dimensions if var_dim and dat_dim parameters are specified
  - **$when** and **$load_parameters** unchanged

status: In CRAN

CST_InsertDim

- A wrapper of **s2dv::InsertDim** for `s2dv_cube` objects
- **Specifications:**
  - It inserts an extra dimension to the **$data** inside the `s2dv_cube` and also adds it to elements: **$dims** and **$coords**
  - The user can provide the values for **$coords[[new_dim]]**, otherwise a sequence of integers from 1 to `lendim` is added, with a warning.
  - A name must be provided

```
> lonlat_temp$exp$dims
dataset  member   sdate   ftime     lat     lon
      1      15       6       3      22      53
> names(lonlat_temp$exp$coords)
[1] "dataset" "member"  "sdate"   "ftime"   "lat"     "lon"
> exp <- CST_InsertDim(data = lonlat_temp$exp, posdim = 2, lendim = 1,
+                      name = "variable", values = c("tas"))
> exp$dims # Check new dimensions and coordinates
dataset variable   member    sdate    ftime      lat      lon
      1        1       15        6        3       22       53
> exp$coords$variable
[1] "tas"
```

status: In CRAN

QuantileMapping allows memb_dim = NULL

- The old function returned **error** if **memb_dim = NULL** when exp_cor was not provided:

```
> res <- QuantileMapping(exp, obs, memb_dim = NULL)
Error in obs[, -sd] : incorrect number of dimensions
```

- Inside the **atomic function** used in multiApply::Apply, member dimension is not subset:

```
.qmapcor <- function(exp, obs, exp_cor = NULL,
                     sdate_dim = 'sdate', ...) {
 # exp: [memb (+ window), sdate]
 # obs: [memb (+ window), sdate]
 # exp_cor: NULL or [memb, sdate]
 if (is.null(exp_cor)) {
   applied <- exp * NA
   for (sd in 1:dim(exp)[sdate_dim]) {
     if (na.rm) {
       # select start date for cross-val
       nas_pos <- which(!is.na(exp[, sd]))
       obs2 <- as.vector(obs[, -sd])
       exp2 <- as.vector(exp[, -sd])
       exp_cor2 <- as.vector(exp[, sd])
    [...]
}
```

target dimensions

status: In CRAN

Comments on Calibration developments

- Changes due to **dat_dim** development
  - The **dat_dim** loop is inside the atomic function **.cal** and it wraps all the calculations
  - Then, for every dataset combination of **exp** (and exp_cor) and **obs**:
    - ➢ If data is **not sufficiently large**, the corresponding values for the dataset combination are **NA** or **exp** (if na.fill = TRUE)
    - ➢ If not, the calibration is computed
- Other changes:
  - If **exp_cor is provided** it will be calibrated: "calibrate forecast instead of hindcast" and `eval.method` will be set as: "hindcast-vs-forecast".

status: In CRAN

CSIndicators

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

New release 1.0.0

- On CRAN: https://cran.r-project.org/web/packages/CSIndicators/
- Installed in workstation & Nord3v2, R/4.1.2-xxx.
- Check the NEWS:
  - Correct vignettes figures links.
  - Exceeding Threshold functions to allow between thresholds or equal threshold options.
  - New `s2dv_cube` object development for all the functions, unit tests, examples and vignettes.

ESS Verification Suite

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Autosubmit + Verification Suite

Autosubmit is a workflow manager developed in-house at BSC-ES. It handles things like job dependencies and the submission and re-submission of jobs to specified HPC platforms. We can also add different 'chunks' (by variable, start date, region…) to the same job.

In the Verification Suite, we use Autosubmit to **perform the same analysis for independent datasets** in a more practical way. Users can now create recipes with multiple **systems**, **start dates**, **variables** and **regions** to be processed independently and in parallel. All the info is in the wiki.

status: in master

Autosubmit + Verification Suite

For example: let's say we want to calibrate a hindcast and compute some metrics using the same methods for two different variables, two different models, and two different start dates.

Now, we can divide this recipe into 8 **atomic recipes,** run all the recipes in parallel on Nord3v2, and retrieve all the results in the same output folder.

```
Analysis:
  Horizon: Seasonal
  Variables:
    - {name: tas, freq: monthly_mean}
    - {name: prlr, freq: monthly_mean}
  Datasets:
    System:
      - {name: ECMWF-SEAS5}
      - {name: Meteo-France-System7}
    Multimodel: no
    Reference:
      - {name: ERA5}
  Time:
    sdate:
      - '0101'
      - '0601'
    fcst_year:
    hcst_start: '2000'
    hcst_end: '2016'
    ftime_min: 1
    ftime_max: 6
  Region:
    - {latmin: -10, latmax: 10, lonmin: -10, lonmax: 10}
```

Autosubmit + Verification Suite

New configuration parameters at the end of the recipe, in the ‘Run’ section:

```
autosubmit: yes
 auto_conf:
   script: /esarchive/scratch/vagudets/repos/auto-s2s/modules/test_parallel_workflow.R
   expid: a5no # if left empty, you will get instructions on how to create a new experiment
   hpc_user: bsc32762 # your hpc username
   wallclock: 04:00 # max. run time for each job in hh:mm
   processors_per_job: 8
   platform: nord3v2
   email_notifications: yes # enable/disable email notifications
   email_address: victoria.agudetse@bsc.es # email address for notifications
   notify_completed: no # notify me by email when a job finishes successfully
   notify_failed: yes # notify me by email when a job fails
```

IMPORTANT: Please save your outputs OUTSIDE of the code directory.

Autosubmit + Verification Suite

Step 1 (only once): Create a new autosubmit experiment. ssh into the autosubmit machine (bscesautosubmit01) and enter the following commands:

```
module load autosubmit/3.14.0-foss-2015a-Python-2.7.9
autosubmit expid -H nord3v2 -d <Description>
```

Step 2: Create your recipe and script.

Step 3: On your workstation or on nord3v2, cd to the code directory and run:

```
source MODULES
Rscript split.R <path_to_your_recipe>
```

This splits the recipe and creates your experiment configuration from a template. Then, follow the instructions that will appear on the terminal.

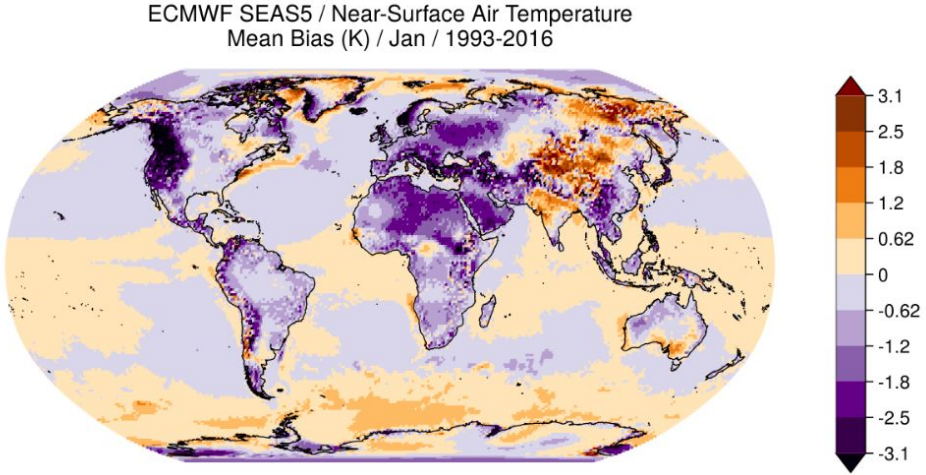
Step 4: You're done! Monitor your experiment from the Autosubmit GUI.

User presentation

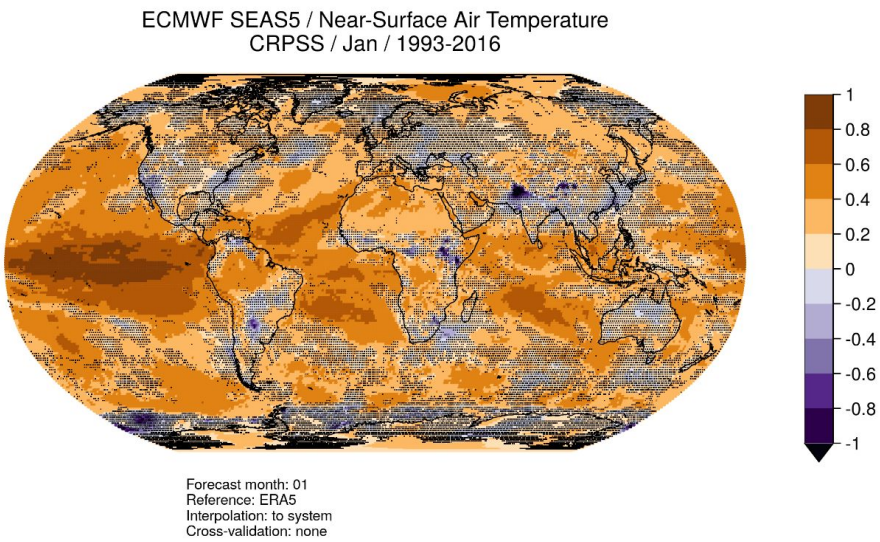
Barcelona Supercomputing Center
Centro Nacional de Supercomputación

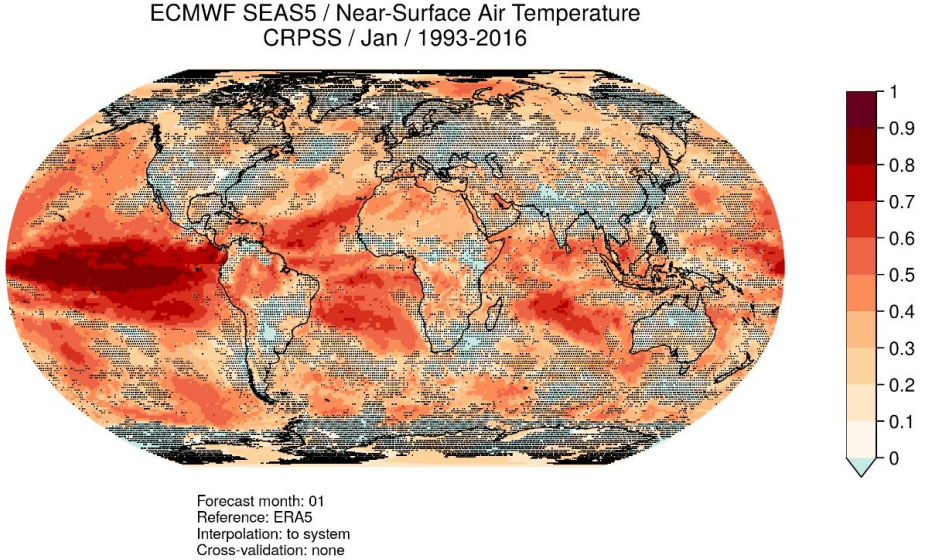
Robinson Projection Plot

```
PlotRobinsonMap <- function(data, lon, lat, original_proj = 4326,
                            target_proj = 54030, brks, cols, colNA = 'green',
                            size = 0.05, coastlines_width = 0.3,
                            dots = NULL, dots_shape = 47, dots_size = 0.005,
                            toptitle = NULL, caption = NULL,
                            fileout = NULL, legend = NULL,
                            lon_dim = 'lon', lat_dim = 'lat',
                            width = 5, height = 4, dpi = 300, units = "in",
                            device = 'png', triangle_ends = NULL,
                            col_inf = NULL, col_sup = NULL,...) {
```

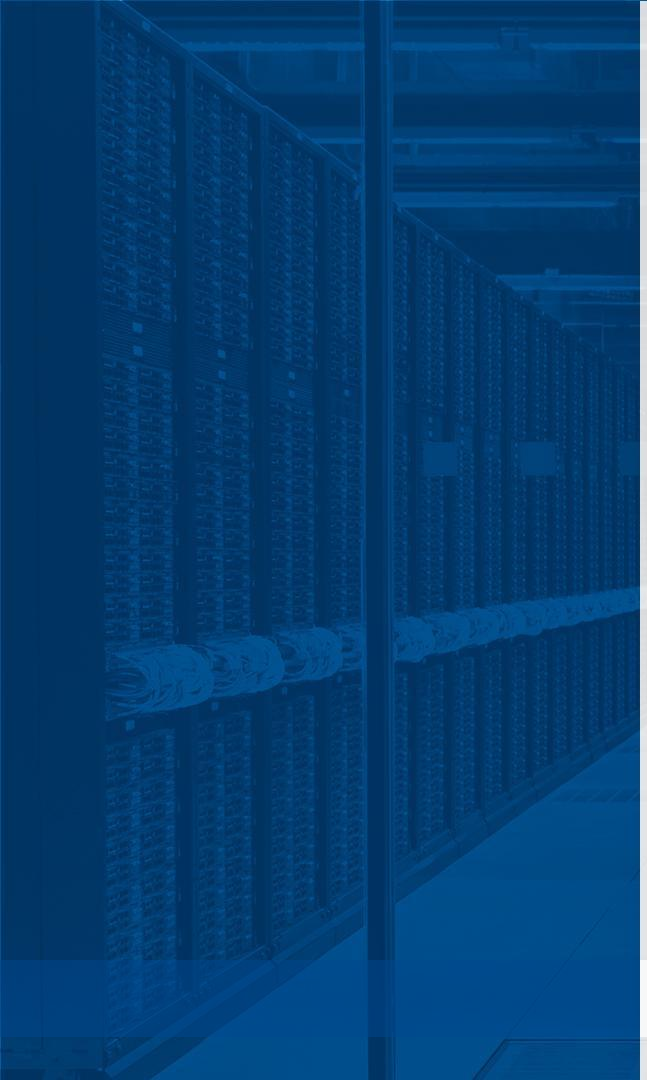


Robinson Projection Plot





Barcelona
Supercomputing
Center
Centro Nacional de Supercomputación



Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Thanks for joining

Next meeting: 4th May, 12h