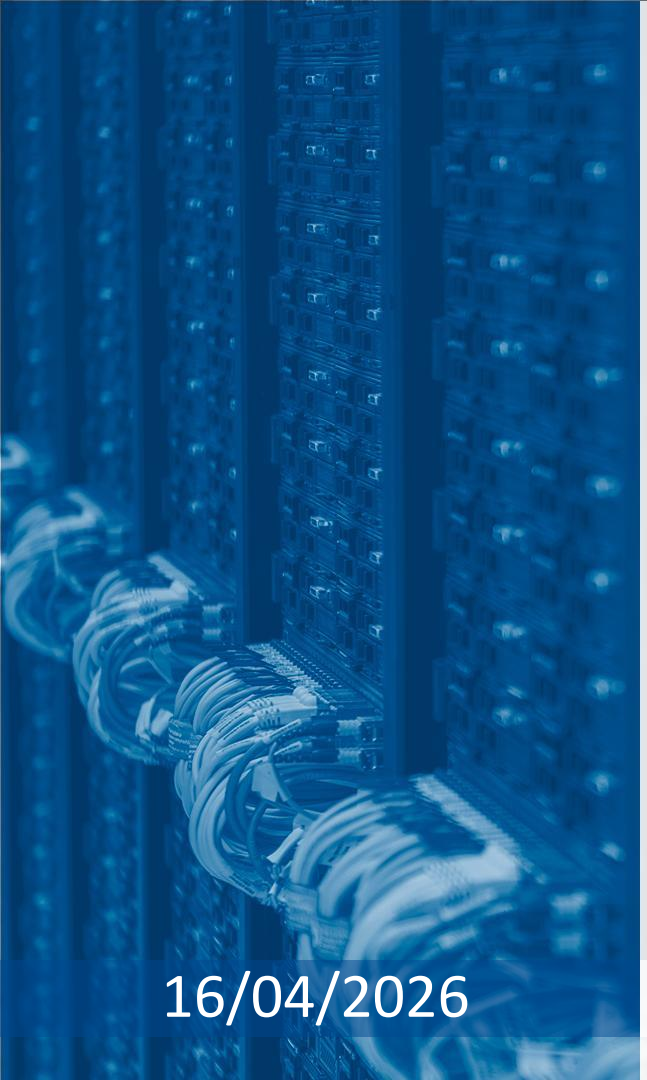


Barcelona Supercomputing Center
Centro Nacional de Supercomputación

R user meeting

16/04/2026

Victòria A., Ariadna B.

Agenda

1. Ice-breaker: dput() function
2. News
    - General
    - startR
    - s2dv
    - CSTools
    - CSIndicators
    - CSDownscale
    - esviz
    - SUNSET
3. User presentation: Aleks Lacima
4. Q&A

Ice-breaker: dput() function

The dput() function

The dput() function takes an object and returns **the R code needed to generate it**.
For example:

```
> (1:4) * pi
[1]  3.141593  6.283185  9.424778 12.566371
> dput((1:4) * pi)
c(3.14159265358979, 6.28318530717959, 9.42477796076938, 12.5663706143592)

> my_list <- list(greeting = "hi")
> my_list
$greeting
[1] "hi”
> dput(my_list)
list(greeting = "hi")
```

The dput() function

This can be useful, for example, to save time and effort when writing unit tests for a function:

```
> exp <- array(rnorm(1000), dim = c(dat = 1, lat = 3, lon = 5, member =
10, sdate = 50))
> obs <- array(rnorm(1000), dim = c(dat = 1, lat = 3, lon = 5, sdate =
50))
> bias <- s2dv::Bias(exp = exp, obs = obs, memb_dim = "member")

# Generate the code to compare results to with dput()
> dput(names(bias))
c("bias", "sign")
> dput(dim(bias$bias))
c(dat = 1L, lat = 3L, lon = 5L)
> dput(bias$sign[1:5])
c(FALSE, FALSE, FALSE, FALSE, FALSE)
```

The dput() function

The `control` parameter offers some flexibility in the format of the output, allowing the user to trade accuracy for simplicity:

```
> dput(dim(bias$bias))
c(dat = 1L, lat = 3L, lon = 5L)
> dput(dim(bias$bias), control = NULL)
c(1, 3, 5)
> dput(dim(bias$bias), control = c("keepInteger"))
c(1L, 3L, 5L)
> dput(dim(bias$bias), control = c("niceNames"))
c(dat = 1, lat = 3, lon = 5)
> dput(dim(bias$bias), control = c("keepInteger", "niceNames"))
c(dat = 1L, lat = 3L, lon = 5L)
```

General

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

New modules on Nord4

We have a new and improved software stack available on Nord4!

The new versions of the R and CDO modules on this machine are:

- ★ `R/4.3.3-foss-2020b`
- ★ `CDO/2.3.0-gompi-2020b`

These modules DO NOT require the use of the Nord3 singularity wrapper!! You can call R and CDO commands normally, **without the "nord3_singu_es" prefix**.

Stamen has installed most of the commonly-used R packages in this new R module. If you find anything is missing, or would like any package to be installed or updated, please open an issue at https://gitlab.earth.bsc.es/es/requests/-/issues/ and tag @smirosla.

startR

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Start(): New parameter ‘transform_method_dim’

startR allows the loading of **multiple variables** into one array, but until now, the same interpolation method was applied to all of them.
The new parameter “**`transform_method_dim`**” allows the user to tell Start() that the interpolation method in `transform_params` **depends on a file dimension** (such as “var”) and provide a list of methods:

```
Start(...,
      var = c('tas', 'prlr'),
      transform = CDORemapper,
      transform_params = list(grid = 'r360x181',
                              method = list(tas = 'bilinear',
                                            prlr = 'conservative')),
      transform_method_dim = 'var',
      ...)
```

issue: https://gitlab.earth.bsc.es/es/startR/-/issues/217
status: in master

Start(): New parameter ‘transform_method_dim’

If the same interpolation method will be used for all the files, **transform_method_dim should be NULL** (default value) and transform_params can have the same format as before:

```
Start(...,
      var = c('tas', 'prlr'),
      transform = CDORemapper,
      transform_params = list(grid = 'r360x181',
                              method = 'bilinear'),
      transform_method_dim = NULL,
      ...)
```

A new use case (Example 1.17) illustrates the usage for this feature.

issue: https://gitlab.earth.bsc.es/es/startR/-/issues/217
status: in master

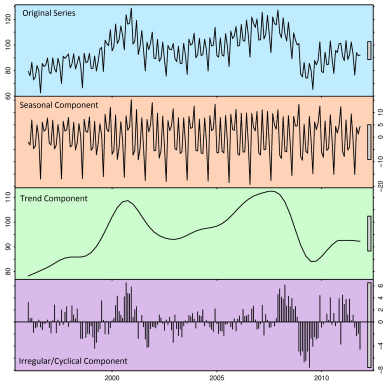
s2dv

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

New function: TimescaleDecomposition()

TimescaleDecomposition():

- Decomposes monthly, daily or yearly grid point time series into:
  - **Trend**: long-term changes
  - **Seasonal**: periodic changes (not available for yearly data)
  - **Decadal**: low-frequency variability
  - **Residual** components: unexplained noise
- Uses LOESS smoothing and Butterworth filtering
- Works over spatial grids and supports additional dimensions (ex: different members or start dates)
- Returns either:
  - Variance percentages of each component
  - Component values across spatial grids
- Developed by Javier Corvillo



For illustrative purposes

New function: TimescaleDecomposition()

Required parameters:

- `data`: A numeric array containing time-series data. The array must include at least the named dimension specified in 'ftime_dim'.

Other parameters:

- `ftime_dim = "time"`
- `fq = 12`
- `lambda = 1`
- `method = "gcv"`
- `return_type = "percentages"`
- `ncores = NULL`

```
# Time-based decomposition
data <- array(rnorm(1000), dim = c(time = 200, lat = 10, lon = 10))

# Add some variability to the data
data <- data +
  array(..., dim = c(200, 10, 10)) + # Trend
  array(..., dim = c(200, 10, 10)) + # Seasonal
  array(..., dim = c(200, 10, 10)) # Decadal
result <- TimescaleDecomposition(data)
```

```
> str(result)
List of 4
 $ pct_trend   : num [1:10, 1:10] 0.6469 0.2744 1.7016 0.2064 ...
 $ pct_seasonal: num [1:10, 1:10] 8.13 2.17 6 6.4 ...
 $ pct_decadal : num [1:10, 1:10] 0.0875 0.0286 0.0456 0.4773 ...
 $ pct_residual: num [1:10, 1:10] 91.1 97.5 92.3 92.9 ...
```

issue: https://gitlab.earth.bsc.es/es/s2dv/-/merge_requests/206
status: in master

CSTools

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

New release CSTools/5.3.1

Version 5.3.1 of CSTools was released on 2026-03-17 and is now available on CRAN and installed on all BSC-ES machines.
The release includes:

- ★ New function CST_ReorderDims() (s2dv_cube wrapper for s2dv::Reorder())
- ★ Several fixes for CST_BindDim() concatenates s2dv_cube objects if a time dimension is selected
- ★ Bugfix in the RainFarm() function

As well as a **change in the maintainer** from Theertha Kariyathan to Victòria Agudetse.

issue: https://gitlab.earth.bsc.es/external/cstools/-/issues/166
status: in production

Improvements in CST_QuantileMapping()

Two issues have been reported about CST_QuantileMapping() in the last few weeks:

1. **The function failed when data contained a full NA grid**, even when setting `na.rm = TRUE`. The bug was fixed by Eren Duzenli

MR: https://gitlab.earth.bsc.es/external/cstools/-/merge_requests/241
status: in master

2. The function fails if package qmap is not explicitly loaded. This causes problems in packages that have CSTools as a dependency. The solution will be included in the master branch soon.

issue: https://gitlab.earth.bsc.es/external/cstools/-/issues/170
status: in branch dev-quantile_mapping_qmap

CST_Calibration(): New parameter ‘window_dim’

New development to implement a **window dimension parameter** window_dim in CST_Calibration(). The functionality is similar to the window_dim parameter already available in CST_QuantileMapping():

“A character string or a vector of character strings, indicating the dimension name(s) in which extra samples are stored. These dimensions are joined to the 'member' dimension. This is useful to correct data, for which robust statistics can be obtained by creating a window of dates around the target date. The default value is NULL”

The development also adjusts some sanity checks in the function to allow more flexibility in the exp and obs dimensions.

issue: https://gitlab.earth.bsc.es/external/cstools/-/issues/170
status: in branch calibration_window, ready to test

CSIndicators

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

New release CSIndicators/1.2.0

Version 1.2.0 of CSIndicators was released on 2026-03-11 and is now available on CRAN and installed on all BSC-ES machines.
The release includes:

- ★ New function HeatIndex() and vignette to compute the Heat Index.
- ★ New function MaxSpellTimeExceedingThreshold() to compute number of consecutive days exceeding a threshold.
- ★ New function DayLength() to compute number of daylight hours
- ★ New function MultiVarExceedingThresholds() to compute binary probabilities based on climate thresholds

As well as a **change in the maintainer** from Theertha Kariyathan to Victòria Agudetse.

issue: https://gitlab.earth.bsc.es/es/csindicators/-/issues/61
status: in production

CSDownscale

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

New release CSDownscale/0.0.2

Version 0.0.2 of CSDownscale was released on 2026-03-05 and is now available on CRAN and installed on all BSC-ES machines.
The release includes:

★ Specification of dependency imports in Interpolation.R

As well as a **change in the maintainer** from Theertha Kariyathan to Victòria Agudetse.

issue: https://gitlab.earth.bsc.es/es/csdownscale/-/issues/15
status: in production

esviz

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

New function: VizPolygonsMap()

- Plot polygon-based spatial data
- Key inputs:
  - **data**: numeric array (region-based)
  - **shapefile**: polygon geometries
  - **region_dim**: links data to shapefile rows
  - [in development] background_shapefile: polygon geometries of the background map

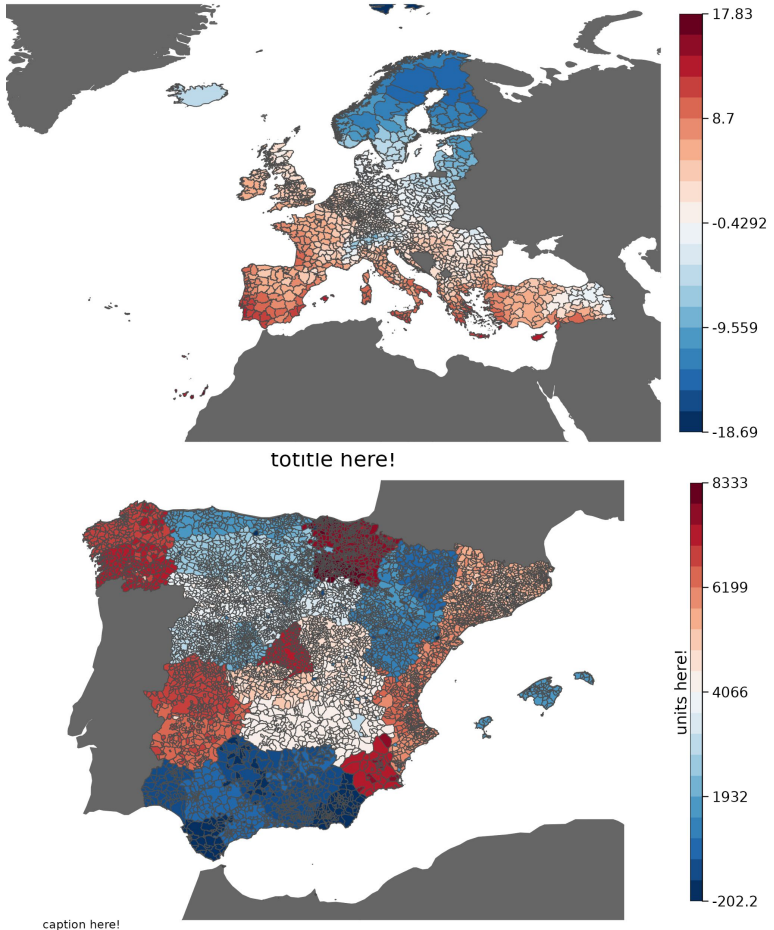
data length must match nrow(shapefile)!

- Flexible map projections: accepts any valid CRS. Default is Robinson.
- Graphical and aesthetic parameters: colorbars, legends, borders, coastlines, ocean/land colors, titles, captions.

MR: https://gitlab.earth.bsc.es/es/esviz/-/merge_requests/44
status: in master

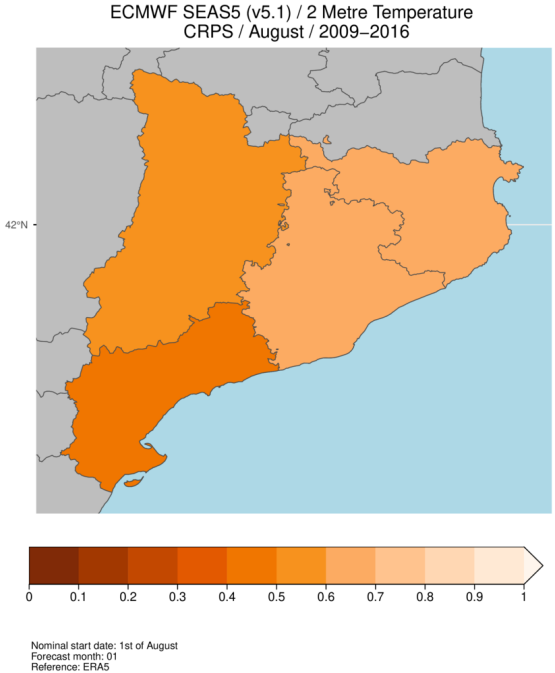
New function: VizPolygonsMap()

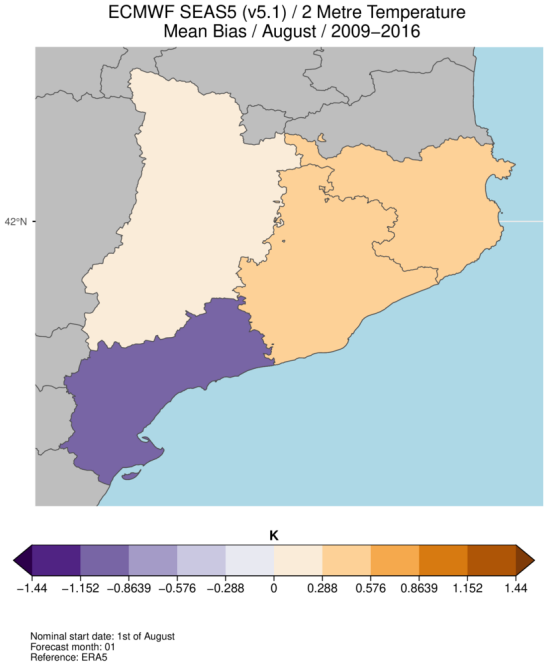
```
VizPolygonsMap(data,shp,
               region_dim = "region",
               target_proj = NULL,
               dots = dots, mask = mask,
               brks = seq(-18,18, by = 4),
               bar_extra_margin = rep(0, 4),
               plot.margin = margin(0,0,0,0, 'in'),
               vertical = TRUE,
               borders = TRUE,
               toptitle = "test toptitle",
               caption = "test caption",
               units = "C",
               box = c(xmin = -37, xmax = 45,
                       ymin = 25, ymax = 75),
               title_scale = 16,
               dot_size = 0.2,
               dot_symbol = 16,
               col_ocean = "white",
               col_coast = "grey30",
               coast_width = 0.3,
               col_land = "grey80",
               col_country_border = NULL,
               shp_border_width = 0.1,
               fileout = '_test.png'),
               width = 8, height = 5,
               size_units = "in", res = 300)
```



New function: VizPolygonsMap()

- In development: integration with SUNSET



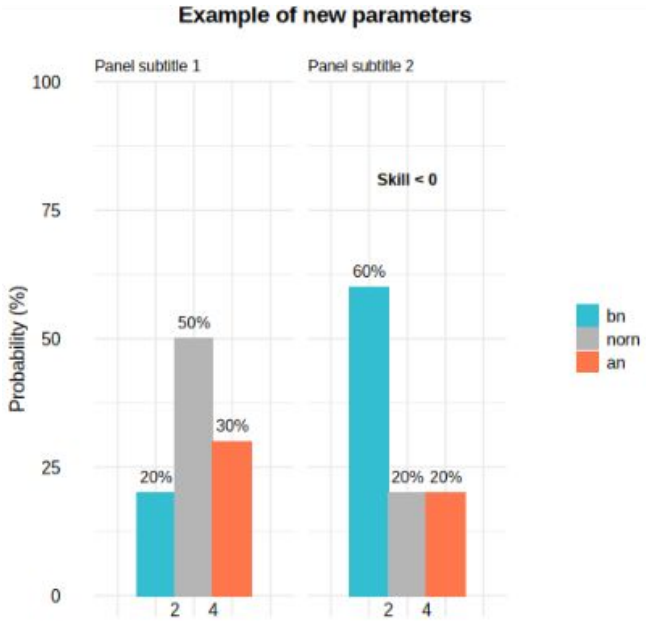


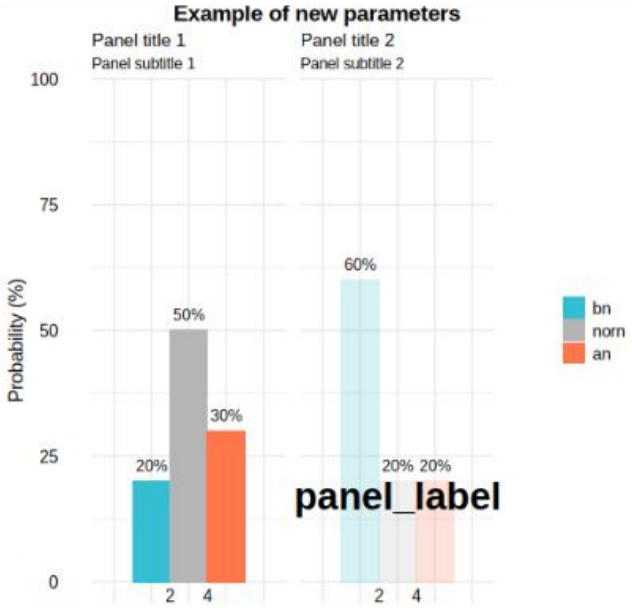
New parameters for BarPlotCat()

Flexibilization of BarPlotCat: the behaviour previously triggered by negative skill imputed through the skill parameter is now customisable via the following parameters:

- `transparency`: Accepts a single value or a vector (one per panel). Allows flexible control of bar transparency.
- `panel_label`: Character vector (or single value) to add text on top of the bars of each panel.
- `label_height`: Controls the y-axis position of the `panel_label`.
- `label_size`: Controls the size of `panel_label`.

New parameters for BarPlotCat()





MR: https://gitlab.earth.bsc.es/es/esviz/-/merge_requests/52
status: in dev-barplotcat_transparency_labels

SUNSET

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Summary of features to be included in 3.2.0

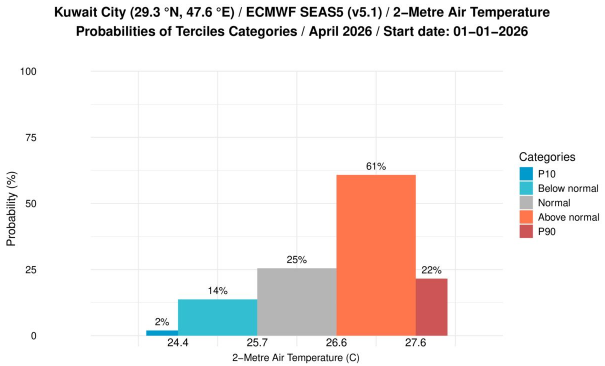
A new release of SUNSET is scheduled for **April 28th, 2026**. This next release will include many useful new features with contributions from many of you. Most notably:

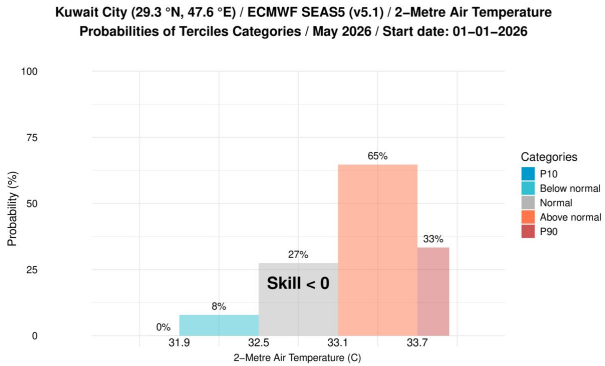
- ★ **New Indicators module** with HeatIndex and Mosquito-borne Disease Suitability indicators.
- ★ **New Blending module** to load loading of backwards requested reference data (and new use case 1.5)
- ★ **Aggregation module:** now allows for **spatial aggregation** using a NetCDF mask
- ★ **Multimodel:** Bugfixes and update of the scripts to work with SUNSET>=3.1.0 and Autosubmit 4
- ★ **Visualization:** introduction of location-based bar plots and increased flexibility of output_conf (region-specific shapefiles and plot aesthetics parameters) and recipe (dots/crosses).

NEWS file: https://gitlab.earth.bsc.es/es/sunset/-/blob/master/NEWS.md
status: in master

Visualization module updates

- Introduction of location-based bar plots:



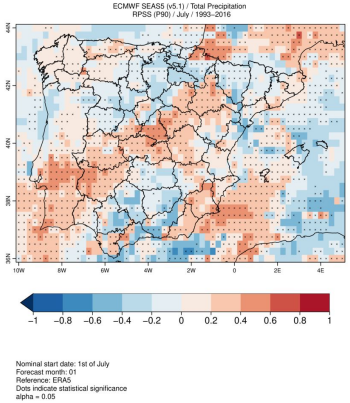


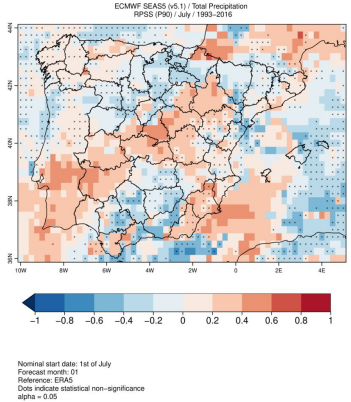
- Graphical configuration parameters in ‘recipe_template.yml’ and ‘output_conf.yml’:
  - New bar_plot recipe parameters:
    - `transparency_no_skill: 0.5`
    - `text_no_skill: "Skill < 0"`
    - `position_text_no_skill: label`

Visualization module updates

- Bugfix: metric RMS now available for plotting
- output_conf:
  - Shapefiles can now be input directly into output_conf → different shapefiles can be passed for different regions! Add: `shapefile: "path_to_shapefile.shp"`
  - New '`caption_height`' parameter: larger values move the caption down
- recipe:
  - New parameters to choose which points to dot (+ caption). Ex: skill_metrics

    `recipe$...$Visualization$dots_on_points_significance <- "significant"/"non-significant"`





Visualization: bar plots

Recipe:

```
    location_plots:
      execute: yes # Whether to execute the location plots
      type: bar_plot # Types of location plots to be included. Options: bar_plot. (str)
      probability_sets:
        # List of probability sets to include in the bar plots.
        # The names in 'probs' and 'extremes' must match the names of probability
        # categories specified in the 'Probabilities' section.
        - set1: {probs: "Terciles", extremes: ["P10", "P90"], category_names: ["Below normal", "Normal", "Above
normal"]}
        # - set2: {probs: "Terciles", category_names: ["Below normal", "Normal", "Above normal"]}
        # - set3: {probs: "Quintiles", category_names: ["Very below normal", "Below normal", "Normal", "Above
normal", "Very above normal"]}
      points:
        # List of point locations.
        - {name: 'Kuwait City', lat: 29.3, lon: 47.6, region: "Kuwait"} # Region name is mandatory and should match
one of the regions defined in the 'Region' section.
        # - {name: 'Barcelona', lat: 40.4, lon: -3.7, region: "Iberia"}
      transparency_no_skill: 0.5 # Transparency to be applied to the bars with negative skill. Ranges from 0 (opaque)
to 1 (transparent). Default is 0.5.
      text_no_skill: "Skill < 0" # Text string to add to the plots with negative skill. Default is "Skill < 0".
      position_text_no_skill: label # Where to position 'text_no_skill'. Can be 'label' (overlaying the bars) or
'subtitle' (plot subtitle). Default is 'label'.
```

User Presentation (Aleks Lacima)

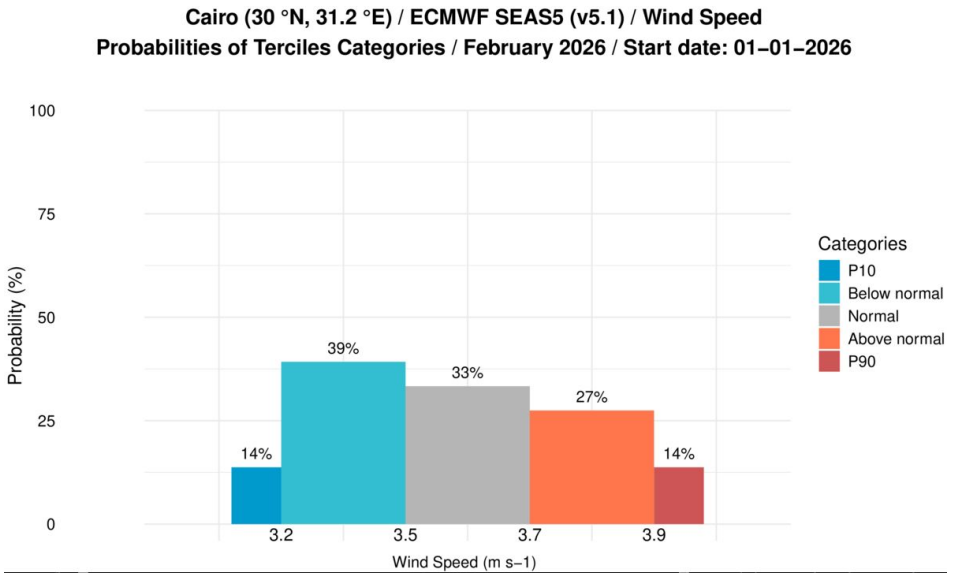
Barcelona Supercomputing Center
Centro Nacional de Supercomputación

New SUNSET visualization: Bar plot

The bar plot is a location-specific forecast visualization that, in its current implementation, displays probabilistic climate predictions for a single grid point.

Specifically, it shows:

- Categorical probabilities (e.g., terciles)
- Extreme probabilities: P10, P90
- Category limits from the observed climatology
- Skill information (e.g., RPSS-based)



New SUNSET visualization: Bar plot

Create a SUNSET recipe that generates bar plot visualizations of categorical forecast probabilities at specific point locations. This use case demonstrates the *location_plots* feature, which extracts forecast information at individual grid points

First, we define the output and code directories, the region and the probabilities:

```
vim use_cases/ex1_6_bar_plot/ex1_6-recipe.yml

```
output_dir: /esarchive/scratch/alacima/data/sunset_dev/test_kuwait_operational
code_dir: /esarchive/scratch/alacima/data/sunset_dev/sunset

Region:
  - {name: "Iberia", latmin: 36, latmax: 44, lonmin: -10, lonmax: 5}

Probabilities:
  percentiles:
    Terciles: [1/3, 2/3]
    P10: [1/10]
    P90: [9/10]
```
```

New SUNSET visualization: Bar plot

The Visualization section now includes both a traditional map plot (*plots*) and point-based location plots (*location_plots*):

New section

```
Visualization:
  plots: most_likely_terciles
  multi_panel: no
  projection: cylindrical_equidistant
  location_plots:
    execute: yes
    type: bar_plot
    probability_sets:
      - set1: {probs: "Terciles", category_names: ["BN", "N", "AN"]}
      - set2: {probs: "Terciles", extremes: ["P10", "P90"], category_names: ["BN", "N", "AN"]}
    points:
      - {name: 'Madrid', lat: 40.4, lon: -3.7, region: 'Iberia'}
      - {name: 'Barcelona', lat: 41.4, lon: 2.2, region: 'Iberia'}
      - {name: 'Lisbon', lat: 38.7, lon: -9.1, region: 'Iberia'}
    transparency_no_skill: 0.5
    text_no_skill: "Skill < 0"
    position_text_no_skill: subtitle
```

New SUNSET visualization: Bar plot

The Visualization section now includes both a traditional map plot (*plots*) and point-based location plots (*location_plots*):

```
Visualization:
  plots: most_likely_terciles
  multi_panel: no
  projection: cylindrical_equidistant
  location_plots:
    execute: yes
    type: bar_plot
    probability_sets:
      - set1: {probs: "Terciles", category_names: ["BN", "N", "AN"]}
      - set2: {probs: "Terciles", extremes: ["P10", "P90"], category_names: ["BN", "N", "AN"]}
    points:
      - {name: 'Madrid', lat: 40.4, lon: -3.7, region: 'Iberia'}
      - {name: 'Barcelona', lat: 41.4, lon: 2.2, region: 'Iberia'}
      - {name: 'Lisbon', lat: 38.7, lon: -9.1, region: 'Iberia'}
    transparency_no_skill: 0.5
    text_no_skill: "Skill < 0"
    position_text_no_skill: subtitle
```

Set to *yes* to run the location_plots block.

New SUNSET visualization: Bar plot

The Visualization section now includes both a traditional map plot (*plots*) and point-based location plots (*location_plots*):

```
Visualization:
  plots: most_likely_terciles
  multi_panel: no
  projection: cylindrical_equidistant
  location_plots:
    execute: yes
    type: bar_plot
    probability_sets:
      - set1: {probs: "Terciles", category_names: ["BN", "N", "AN"]}
      - set2: {probs: "Terciles", extremes: ["P10", "P90"], category_names: ["BN", "N", "AN"]}
    points:
      - {name: 'Madrid', lat: 40.4, lon: -3.7, region: 'Iberia'}
      - {name: 'Barcelona', lat: 41.4, lon: 2.2, region: 'Iberia'}
      - {name: 'Lisbon', lat: 38.7, lon: -9.1, region: 'Iberia'}
    transparency_no_skill: 0.5
    text_no_skill: "Skill < 0"
    position_text_no_skill: subtitle
```

Type of plot, only *bar_plot* is currently implemented.

New SUNSET visualization: Bar plot

The Visualization section now includes both a traditional map plot (*plots*) and point-based location plots (*location_plots*):

```
Visualization:
  plots: most_likely_terciles
  multi_panel: no
  projection: cylindrical_equidistant
  location_plots:
    execute: yes
    type: bar_plot
    probability_sets:
      - set1: {probs: "Terciles", category_names: ["BN", "N", "AN"]}
      - set2: {probs: "Terciles", extremes: ["P10", "P90"], category_names: ["BN", "N", "AN"]}
    points:
      - {name: 'Madrid', lat: 40.4, lon: -3.7, region: 'Iberia'}
      - {name: 'Barcelona', lat: 41.4, lon: 2.2, region: 'Iberia'}
      - {name: 'Lisbon', lat: 38.7, lon: -9.1, region: 'Iberia'}
    transparency_no_skill: 0.5
    text_no_skill: "Skill < 0"
    position_text_no_skill: subtitle
```

Probability sets to plot. You can define as many as you want. This will be reflected in the figure filename.

New SUNSET visualization: Bar plot

The Visualization section now includes both a traditional map plot (*plots*) and point-based location plots (*location_plots*):

```
Visualization:
  plots: most_likely_terciles
  multi_panel: no
  projection: cylindrical_equidistant
  location_plots:
    execute: yes
    type: bar_plot
    probability_sets:
      - set1: {probs: "Terciles", category_names: ["BN", "N",
      - set2: {probs: "Terciles", extremes: ["P10", "P90"], ca
    points:
      - {name: 'Madrid', lat: 40.4, lon: -3.7, region: 'Iberia'}
      - {name: 'Barcelona', lat: 41.4, lon: 2.2, region: 'Iberia'}
      - {name: 'Lisbon', lat: 38.7, lon: -9.1, region: 'Iberia'}
    transparency_no_skill: 0.5
    text_no_skill: "Skill < 0"
    position_text_no_skill: subtitle
```

Locations to plot, specified through a latitude and longitude coordinate The name is sensitive to upper case letters and special characters. The region also needs to be specified.

New SUNSET visualization: Bar plot

The Visualization section now includes both a traditional map plot (*plots*) and point-based location plots (*location_plots*):

```
Visualization:
  plots: most_likely_terciles
  multi_panel: no
  projection: cylindrical_equidistant
  location_plots:
    execute: yes
    type: bar_plot
    probability_sets:
      - set1: {probs: "Terciles", category_names: ["BN", "N", "AN"]}
      - set2: {probs: "Terciles", extremes: ["P10", "P90"], category_names: ["BN", "N", "AN"]}
    points:
      - {name: 'Madrid', lat: 40.4, lon: -3.7, region: 'Iberi
      - {name: 'Barcelona', lat: 41.4, lon: 2.2, region: 'Ibe
      - {name: 'Lisbon', lat: 38.7, lon: -9.1, region: 'Iberi
    transparency_no_skill: 0.5
    text_no_skill: "Skill < 0"
    position_text_no_skill: subtitle
```

Parameters for absence of skill:

- Transparency of the bars
- Text to be displayed
- Position of the text

New SUNSET visualization: Bar plot

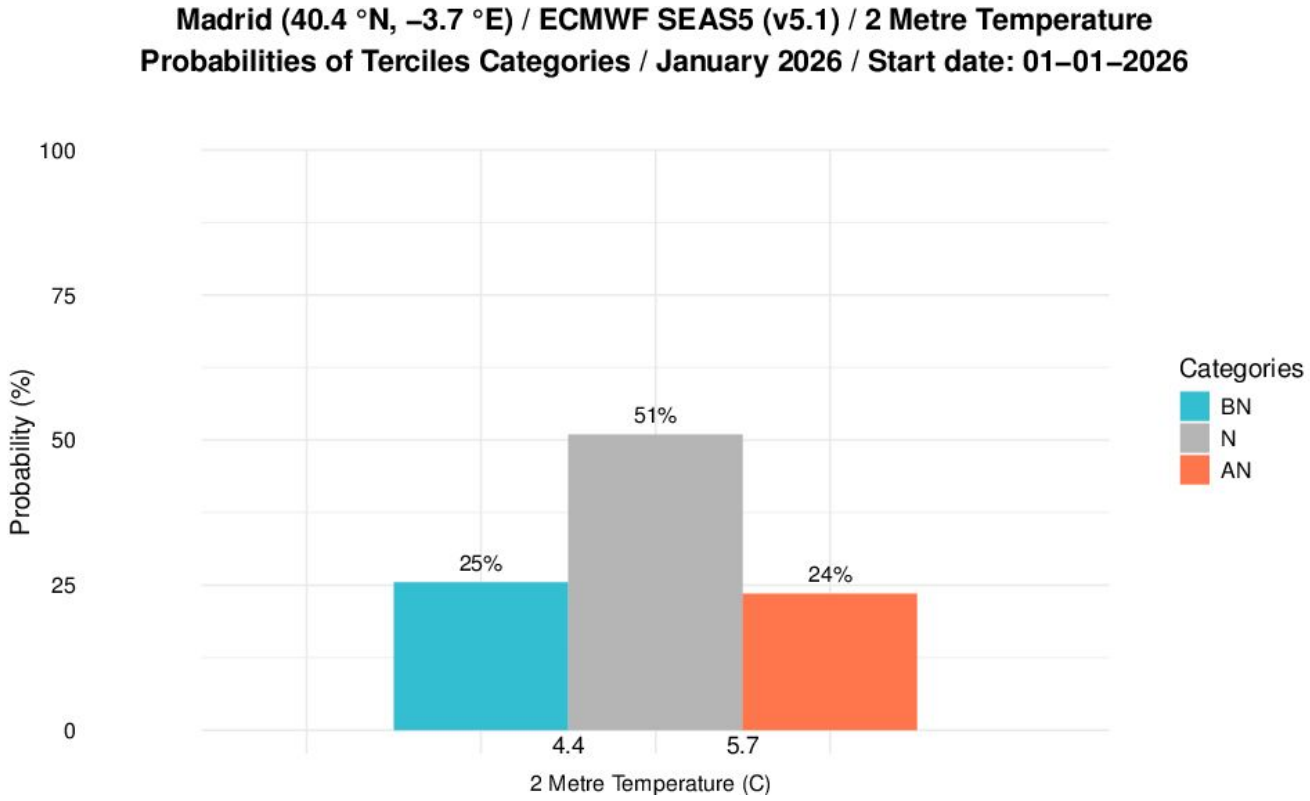
We have asked for two variables, three forecast times, two probability sets and three locations. Can you guess how many bar plots will be generated?

```
Analysis:
  Horizon: seasonal
  Variables:
    name: tas, prlr
    freq: monthly_mean
    units: C, mm
  Datasets:
    System:
      name: ECMWF-SEAS5.1
    Multimodel:
      execute: no
    Reference:
      name: ERA5
  Time:
    sdate: '0101'
    fcst_year: 2026
    hcst_start: '1993'
    hcst_end: '2016'
    ftime_min: 1
    ftime_max: 3
```

```
forecast_bar_tercile_categories-20260101_barcelona_set1_rpss_ft01.pdf
forecast_bar_tercile_categories-20260101_barcelona_set1_rpss_ft02.pdf
forecast_bar_tercile_categories-20260101_barcelona_set1_rpss_ft03.pdf
forecast_bar_tercile_categories-20260101_barcelona_set2_rpss_ft01.pdf
forecast_bar_tercile_categories-20260101_barcelona_set2_rpss_ft02.pdf
forecast_bar_tercile_categories-20260101_barcelona_set2_rpss_ft03.pdf
forecast_bar_tercile_categories-20260101_lisbon_set1_rpss_ft01.pdf
forecast_bar_tercile_categories-20260101_lisbon_set1_rpss_ft02.pdf
forecast_bar_tercile_categories-20260101_lisbon_set1_rpss_ft03.pdf
forecast_bar_tercile_categories-20260101_lisbon_set2_rpss_ft01.pdf
forecast_bar_tercile_categories-20260101_lisbon_set2_rpss_ft02.pdf
forecast_bar_tercile_categories-20260101_lisbon_set2_rpss_ft03.pdf
forecast_bar_tercile_categories-20260101_madrid_set1_rpss_ft01.pdf
forecast_bar_tercile_categories-20260101_madrid_set1_rpss_ft02.pdf
forecast_bar_tercile_categories-20260101_madrid_set1_rpss_ft03.pdf
forecast_bar_tercile_categories-20260101_madrid_set2_rpss_ft01.pdf
forecast_bar_tercile_categories-20260101_madrid_set2_rpss_ft02.pdf
forecast_bar_tercile_categories-20260101_madrid_set2_rpss_ft03.pdf
forecast_most_likely_tercile-20260101_ft01.pdf
forecast_most_likely_tercile-20260101_ft02.pdf
forecast_most_likely_tercile-20260101_ft03.pdf
```

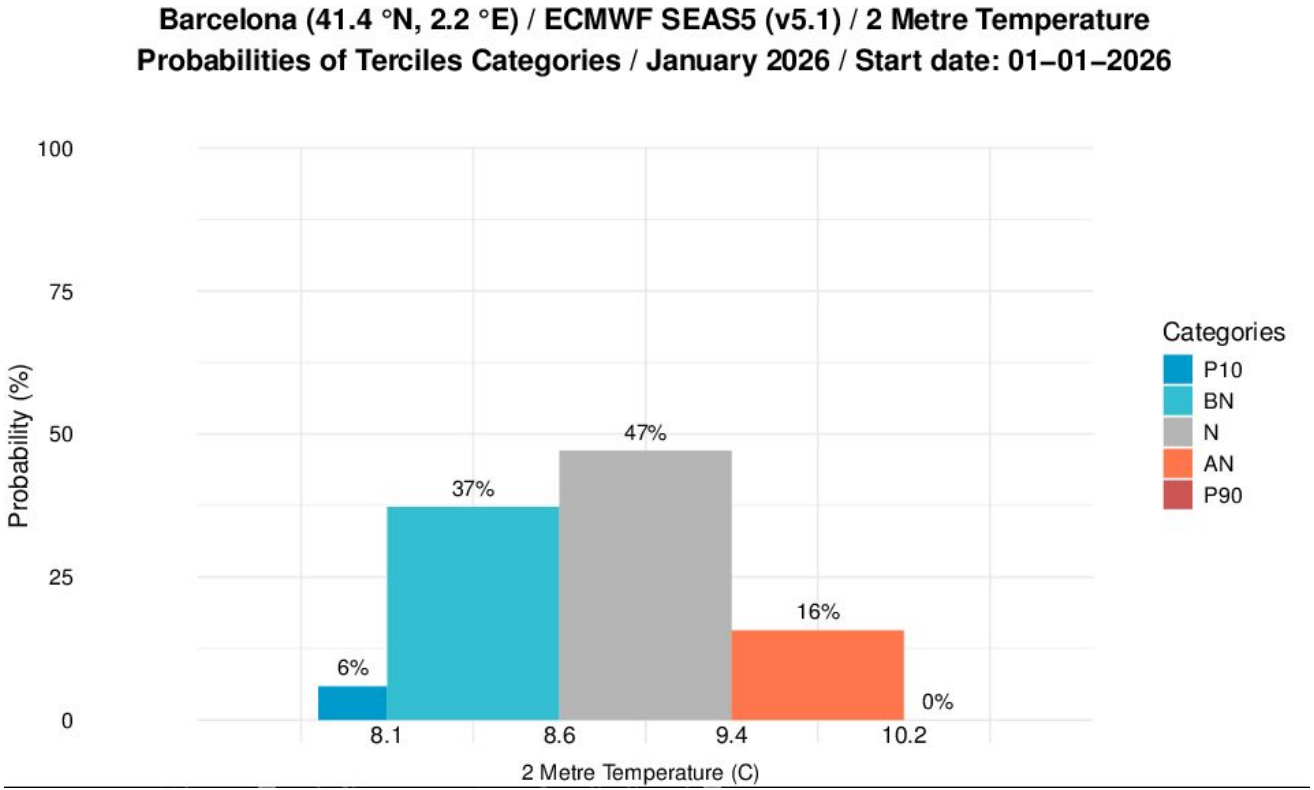
New SUNSET visualization: Bar plot

Finally, we can take a look at the plots:



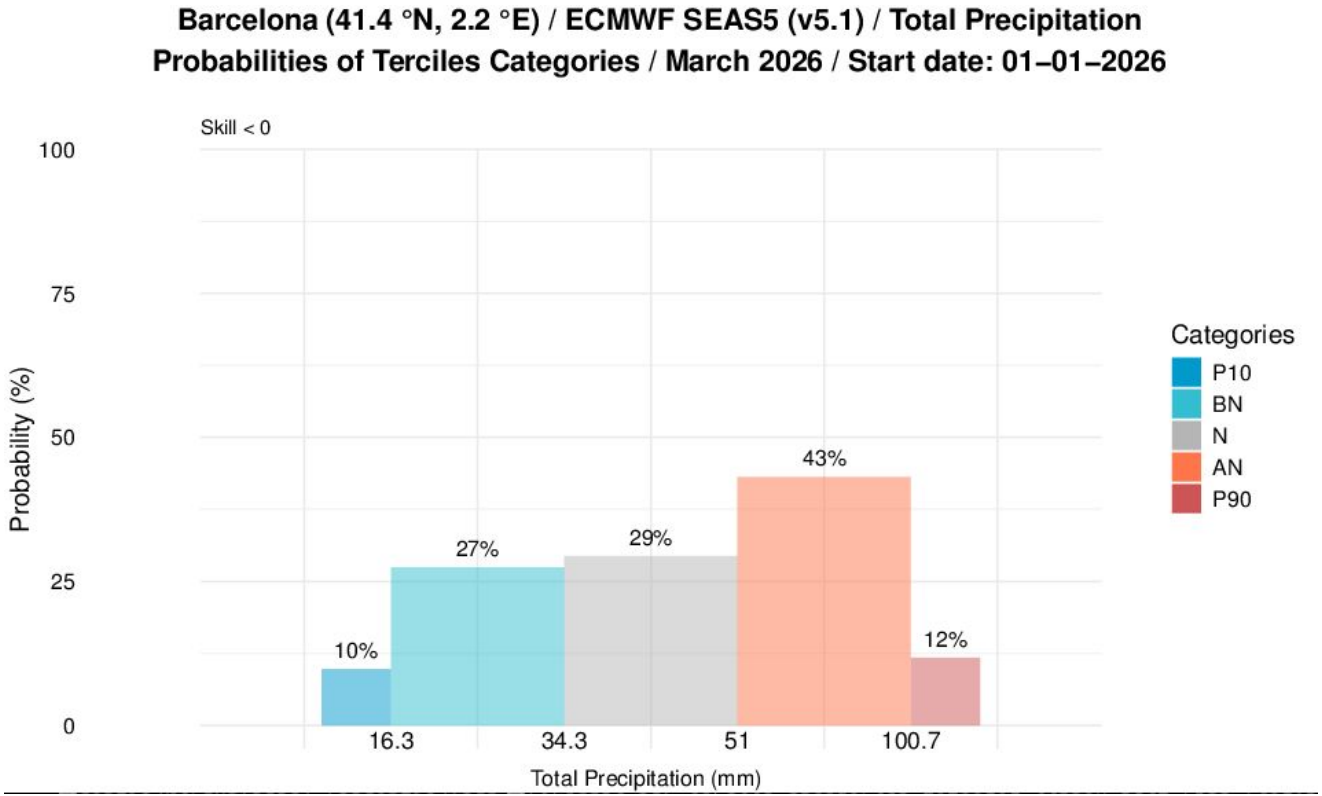
New SUNSET visualization: Bar plot

Finally, we can take a look at the plots:



New SUNSET visualization: Bar plot

Finally, we can take a look at the plots:



Thanks for joining

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

Next meeting: June 4th