



**Barcelona
Supercomputing
Center**

Centro Nacional de Supercomputación

R user meeting

12/09/2024

Victòria Agudetse, Ariadna Batalla

Agenda

1. Ice-breaker:
2. News
 - General R
 - startR
 - s2dv
 - esviz
 - SUNSET
3. Presentation: Indicators module for SUNSET (Alba)
4. Q&A

Ice-breaker:

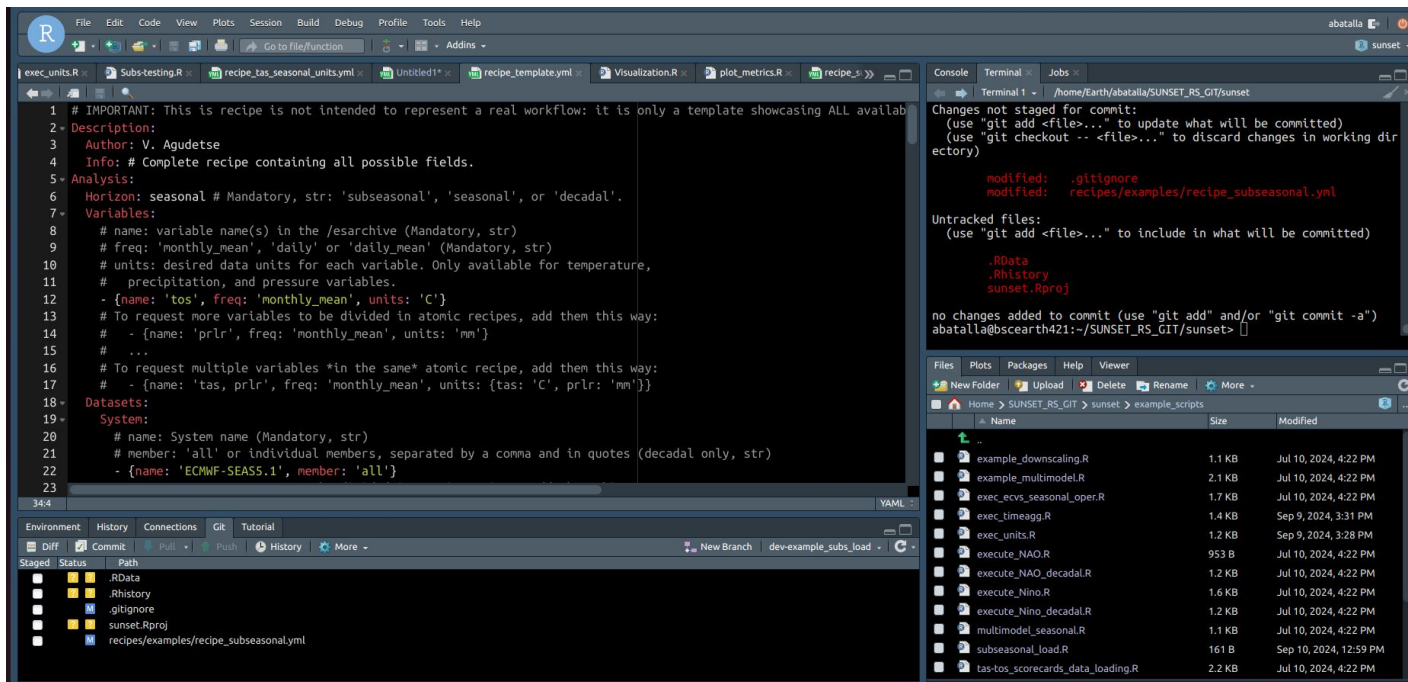


**Barcelona
Supercomputing
Center**

Centro Nacional de Supercomputación

Git in RStudio

Brief overview of how to use Git in RStudio



The screenshot displays the RStudio interface with three main panels:

- Code Editor:** Shows a recipe file with the following content:

```
1 # IMPORTANT: This is recipe is not intended to represent a real workflow: it is only a template showcasing ALL available
2 Description:
3 Author: V. Agudetse
4 Info: # Complete recipe containing all possible fields.
5 Analysis:
6 Horizon: seasonal # Mandatory, str: 'subseasonal', 'seasonal', or 'decadal'.
7 Variables:
8 # name: variable name(s) in the /esarchive (Mandatory, str)
9 # freq: 'monthly_mean', 'daily' or 'daily_mean' (Mandatory, str)
10 # units: desired data units for each variable. Only available for temperature,
11 # precipitation, and pressure variables.
12 - {name: 'tos', freq: 'monthly_mean', units: 'C'}
13 # To request more variables to be divided in atomic recipes, add them this way:
14 # - {name: 'prlr', freq: 'monthly_mean', units: 'mm'}
15 # ...
16 # To request multiple variables *in the same* atomic recipe, add them this way:
17 # - {name: 'tas, prlr', freq: 'monthly_mean', units: {'tas': 'C', prlr: 'mm'}}
18 Datasets:
19 System:
20 # name: System name (Mandatory, str)
21 # member: 'all' or individual members, separated by a comma and in quotes (decadal only, str)
22 - {name: 'ECMWF-SEASS.1', member: 'all'}
23
```
- Terminal:** Shows the output of a `git status` command:

```
Changes not staged for commit:
(use "git add <file>..." to update what will be committed)
(use "git checkout -- <file>..." to discard changes in working directory)

modified:   .gitignore
modified:   recipes/examples/recipe_subseasonal.yml

Untracked files:
(use "git add <file>..." to include in what will be committed)

.RData
.Rhistory
sunset.Rproj

no changes added to commit (use "git add" and/or "git commit -a")
abatalla@bscearth421:~/SUNSET_RS_GIT/sunset>
```
- File Explorer:** Shows a directory listing of files in the `example_scripts` folder:

Name	Size	Modified
example_downscaling.R	1.1 KB	Jul 10, 2024, 4:22 PM
example_multimodel.R	2.1 KB	Jul 10, 2024, 4:22 PM
exec_ecvs_seasonal_oper.R	1.7 KB	Jul 10, 2024, 4:22 PM
exec_timeagg.R	1.4 KB	Sep 9, 2024, 3:31 PM
exec_units.R	1.2 KB	Sep 9, 2024, 3:28 PM
execute_NAO.R	953 B	Jul 10, 2024, 4:22 PM
execute_NAO_decadal.R	1.2 KB	Jul 10, 2024, 4:22 PM
execute_Nino.R	1.6 KB	Jul 10, 2024, 4:22 PM
execute_Nino_decadal.R	1.2 KB	Jul 10, 2024, 4:22 PM
multimodel_seasonal.R	1.1 KB	Jul 10, 2024, 4:22 PM
subseasonal_load.R	161 B	Sep 10, 2024, 12:59 PM
tas-tos_scorecards_data_loading.R	2.2 KB	Jul 10, 2024, 4:22 PM

General R



**Barcelona
Supercomputing
Center**

Centro Nacional de Supercomputación

Code style guide for contributions

We have compiled an R style guide for contributions to the R packages and SUNSET. It is a less strict version of the tidyverse guide.

This will make reviewing easier and will ensure we have readable and consistent code across all the functions.

You can find it here, along with some tips:

https://earth.bsc.es/wiki/doku.php?id=tools:style_guides:r

Here are some of the most important points:

- ★ Add **comments** to your code to make it easier to read and understand
- ★ Use **two spaces** for indentation
- ★ Use **'<-'** and **not '='** for variable assignment
- ★ Include **spaces after** commas, for/if/while, closing parentheses `)` and operators
- ★ Try to limit line length to **80 characters** when possible



startR



**Barcelona
Supercomputing
Center**

Centro Nacional de Supercomputación

New release of startR

The new version of startR, v2.4.0, is being installed today on all machines. It includes the following changes:

- ★ Allow chunking along inner dimensions that go across file dimensions
- ★ Allow more than one file dimension to be specified in "metadata_dims"
- ★ Add check and warning for when special wildcard "\$var\$" is missing in the path
- ★ Bugfix: Start() retrieve correct time steps when time is across file dimension and the time steps of the first files are skipped
- ★ Bugfix: Generate correct file paths when a file dimension has multiple depending dimensions

It is now installed on all machines. If you encounter any issues, please report them to us!

Improvement in the flexibility of metadata retrieval

By default, Start() only retrieves the variable metadata from the **first file** it reads.

With the parameter metadata_dims we can specify extra file dimensions along which to look for the metadata. For example it is useful when:

- A) **More than one variable is requested** (metadata_dims = "var"). See: https://earth.bsc.es/gitlab/es/startR/-/blob/master/inst/doc/faq.md#20-use-metadata_dims-to-retrieve-variable-metadata
- B) **The first file is missing**: in this case specifying another file dimension as metadata_dims will retrieve the variable metadata for all the files. See: <https://earth.bsc.es/gitlab/es/startR/-/blob/master/inst/doc/faq.md#19-get-metadata-when-the-first-file-does-not-exist>

status: in master

issue: <https://earth.bsc.es/gitlab/es/startR/-/issues/203>

Improvement in the flexibility of metadata retrieval

Until now, only one file dimension could be specified, so both cases could not be combined. Now it is possible to specify more than one file dimension, to make sure the metadata is retrieved correctly in the case **A + B**.

```
hcst.path <- "/esarchive/exp/ncp/cfs-v2/weekly_mean/s2s/$var$_f24h/$var$_$file_date$.nc"
file_date <- c("19990711", "19990715")
variable <- c("tas", "prlr")

data <- Start(dat = hcst.path,
             var = variable,
             file_date = file_date,
             ...
             metadata_dims = c('var', 'file_date'),
             return_vars = list(latitude = 'dat',
                                longitude = 'dat',
                                time = 'file_date'),
             retrieve = FALSE)
```

status: in master

issue: <https://earth.bsc.es/gitlab/es/startR/-/issues/203>

New sanity check for wildcard '\$var\$'

The “var” dimension is a special file dimension that is required by startR in order to retrieve the data, and the “\$var\$” wildcard is required in the path.

```
path <- "/esarchive/exp/ncep/cfs-v2/weekly_mean/s2s/$var$_f24h/$var$_$file_date$.nc" # OK!  
path <- "/esarchive/exp/ncep/cfs-v2/weekly_mean/s2s/tas_f24h/tas_$file_date$.nc" # Wrong!
```

Sometimes Start() works even if “\$var\$” is not included, but in other cases it fails with strange errors. We have included a **check that raises a warning** if “\$var\$” is not found in the path:

```
"The special wildcard '$var$' is not present in the file. This might cause  
Start() to fail if it cannot parse the inner dimensions in all the files."
```

status: in master

MR: https://earth.bsc.es/gitlab/es/startR/-/merge_requests/233

s2dv



**Barcelona
Supercomputing
Center**

Centro Nacional de Supercomputación

Fair RPS() and RPSS(): Allow probabilities as input

Instead of the experiment and observation arrays, RPS() and RPSS() can accept the exp and obs probabilities as the input arrays directly. To compute the fair version of these metrics (Fair = TRUE), new parameters are required:

- nmemb: A numeric value indicating the number of members used to compute the probabilities. Default value is NULL.
- nmemb_ref (RPSS): A numeric value indicating the number of members of the reference forecast 'ref'. If 'ref' is a climatology, nmemb_ref should be the number of years used to compute the climatology.

Issue: <https://earth.bsc.es/gitlab/es/s2dv/-/issues/119>

status: in branch fairrps

CRPS() and CRPSS(): Add parameter na.rm

This development is to include the na.rm parameter to CRPS() and CRPSS(), with three possibilities:

- **TRUE:** The NA values are removed along the start date dimension before computing the CRPS(S).
- **FALSE:** NA is returned if any NA values are present in the data point.
- **A number from 0 to 1:** The maximum fraction of NAs allowed. If the fraction of NAs in the data point is lower than na.rm, the CRPS(S) will be computed. If it is higher, the result will be NA.

Issue: <https://earth.bsc.es/gitlab/es/s2dv/-/issues/116>

status: in branch dev-crpss

MSSS() and RMSSS(): Use climatology as reference

The MSSS() and RMSSS() functions accept a user-provided reference dataset through the parameter `ref`.

Currently, if `ref = NULL` (default option), an array filled with zeros is used as the reference. This **assumes that the climatology has been previously removed** from the `exp` and `obs` datasets. If this is not the case, the result will be incorrect.

A new development is in the works to compute the climatology when `ref` is `NULL`.

MR: https://earth.bsc.es/gitlab/es/s2dv/-/merge_requests/190

status: in branch `dev_rmsss_ref`

CSIndicators



**Barcelona
Supercomputing
Center**

Centro Nacional de Supercomputación

CST_PeriodStandardization()

The bugfix for the `dates` parameter has been tested and merged. Previously, the `dates` parameter was not passed to the internal `PeriodStandardization` function, which prevented the correct use of the parameter `ref_period`. With the inclusion of `dates`, `ref_period` now works as intended.

```
CST_PeriodStandardization <- function(...) {  
  ...  
  res <- PeriodStandardization(data = data$data, data_cor = data_cor$data,  
                               dates = data$attrs$Dates, ...)  
  ...  
}
```

Additionally, fix of grammatical errors in the documentation and in a warning.

MR: https://earth.bsc.es/gitlab/es/csindicators/-/merge_requests/66

status: in master



esviz



**Barcelona
Supercomputing
Center**

Centro Nacional de Supercomputación

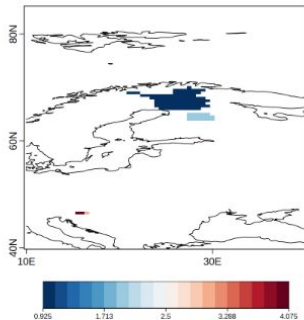
shapeToMask: vignette

```
[7]: # Open the NetCDF file
nc_file <- nc_open("../mask_area_false.nc")

# Extract variables
lat <- ncvar_get(nc_file, "lat")
lon <- ncvar_get(nc_file, "lon")
vari_1 <- ncvar_get(nc_file, "vari_1")

vari_1[vari_1 == 0] <- NA

# Do the prints
s2dv::PlotEquiMap(var = vari_1,
  lat = lat,
  lon = lon,
  filled.continents = FALSE,
  colNA = 'white',
  color_fun = clim.palette(palette = "bluered"),
  # boxlim = c(11, 85, 40, 40)
)
```

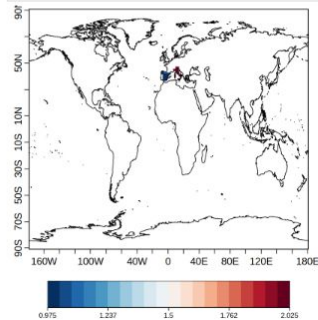


```
[13]: # Open the NetCDF file
nc_file <- nc_open("../mask_shape_gadm.nc")

# Extract variables
lat <- ncvar_get(nc_file, "lat")
lon <- ncvar_get(nc_file, "lon")
vari_1 <- ncvar_get(nc_file, "vari_1")

vari_1[vari_1 == 0] <- NA

# Do the prints
s2dv::PlotEquiMap(var = vari_1,
  lat = lat,
  lon = lon,
  filled.continents = FALSE,
  colNA = 'white',
  color_fun = clim.palette(palette = "bluered"),
  # boxlim = c(11, 85, 40, 40)
)
```



URL:

https://earth.bsc.es/gitlab/es/esviz/-/blob/develop-ShapeToMask_area/vignettes/shape_to_mask.ipynb
status: in develop-ShapeToMask_area

SUNSET



**Barcelona
Supercomputing
Center**

Centro Nacional de Supercomputación

Subseasonal data: loading and visualization

The **loading** and **visualization** modules can now process subseasonal data. The code to load and visualize the data is the same as for seasonal data. The intermediate steps are up to the user.

Example for **loading** subseasonal data at

https://earth.bsc.es/gitlab/es/sunset/-/blob/master/example_scripts/subseasonal_load.R:

```
source("modules/Loading/Loading.R")

recipe_file <- "recipes/examples/recipe_subseasonal.yml"
recipe <- prepare_outputs(recipe_file)
data <- Loading(recipe)
```

Merge request: https://earth.bsc.es/gitlab/es/sunset/-/merge_requests/145

status: in master

Subseasonal data: loading

Recipe template showcasing all options. In subseasonal the time steps are weeks.

```
Time:
  sdate: 20240711 # %Y%m%d
         # Start date (Mandatory, int)
         # For Subseasonal, there are two options:
         # - 'YYYYmmdd' (e.g. 20240104); A specific date of the year
         # - 'YYYY' (e.g. 2024); A year to evaluate all 52 weeks initialized
         #   on week_day. This will divide the recipe into 52 atomic recipes.
  fcst_year: '2024' # Forecast initialization year 'YYYY' (Optional, int)
  # For subseasonal, a specific date should be requested and it should be
  # the same as the one defined as sdate (to be coherent between
  # forecast provision and assessment.
  hcst_start: '1999' # 'YYYY' (Mandatory, int)
  hcst_end: '2006' # 'YYYY' (Mandatory, int)
  ftime_min: 1 # First forecast time step in weeks. Starts at "1". (Mandatory, int)
  ftime_max: 4 # Last forecast time step in weeks. Starts at "1". (Mandatory, int)
  # For subseasonal, there are three extra parameters:
  week_day: Thursday # currently only available for Thursday (Subseasonal only, str)
  sday_window: 3 # The number of days use for calibration (Subseasonal only, int)
  sweek_window: 3 # The number of weeks to use for assessment (Subseasonal only, int)
```

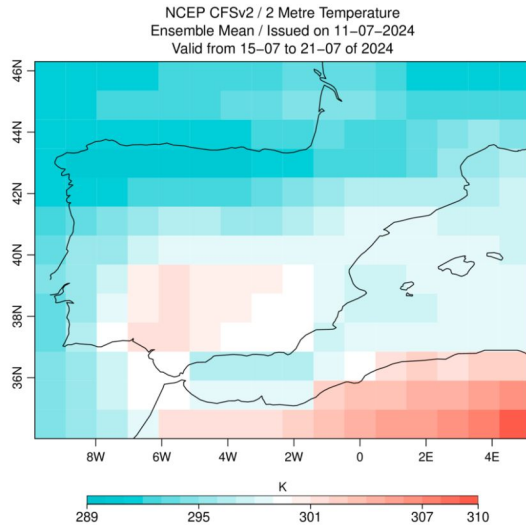
2 options

same as sdate

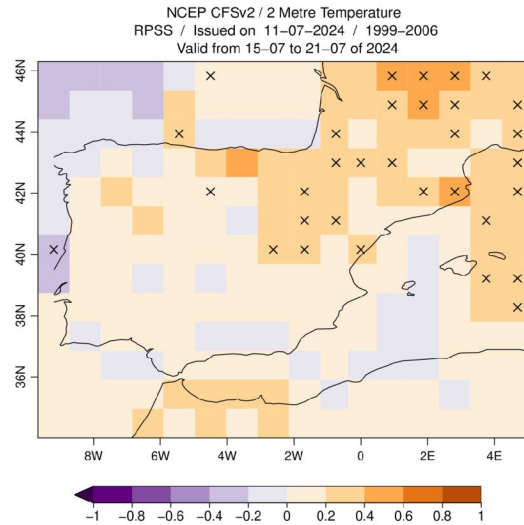
3 new parameters

Subseasonal data: visualization

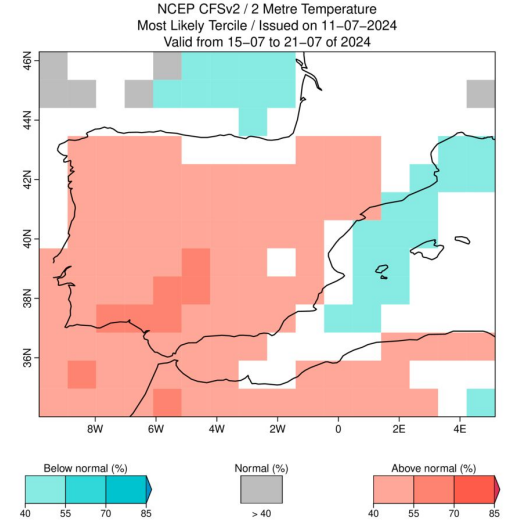
Forecast ensemble mean:



Skill:



Most likely terciles:



Merge request: https://earth.bsc.es/gitlab/es/sunset/-/merge_requests/134 (see examples)

status: in master

Access to GPFS data

Some of the seasonal and decadal models in esarchive are present in GPFS because the data in esarchive cannot be accessed from CTE-AMD or MN5.

The option filesystem = 'gpfs' has been added to SUNSET to be able to load the available datasets from these machines.

```
Run:  
  Loglevel: INFO  
  Terminal: yes  
  filesystem: gpfs
```

If any additional datasets are added to GPFS and you would like to access them with SUNSET, please let us know.

Issue: <https://earth.bsc.es/gitlab/es/sunset/-/issues/125>

status: in master

Refactoring of the Scorecards module

The significance computation for the Mean Bias and Spread-to-error ratio, as well as the refactoring of the Scorecards code have been included in the master branch.

A reminder that the `Scorecards_calculations()` function needs to be called in the script now to do the preliminary calculations for the Scorecards.

```
(...)  
# Compute skill metrics  
skill_metrics <- Skill(recipe, data)  
# Compute statistics  
statistics <- Statistics(recipe, data)  
# Pre-computations required for the scorecards  
Scorecards_calculations(recipe, data = data,  
                         skill_metrics = skill_metrics,  
                         statistics = statistics)
```

MR: https://earth.bsc.es/gitlab/es/sunset/-/merge_requests/141

status: in master

User presentation



**Barcelona
Supercomputing
Center**

Centro Nacional de Supercomputación

SUNSET: module Indicators

Drought indicators:

- SPEI (Standardized Precipitation-Evapotranspiration Index)
- SPI (Standardized Precipitation Index)



[CST_PeriodPET](#)
[CST_PeriodAccumulation](#)
[CST_PeriodStandardization](#)

Indicators:

SPEI:

```
return_spei: yes # yes/no
PET_method: hargreaves # options: none, hargreaves, hargreaves_modified, thornthwaite
Nmonths_accum: 3 # any integer covered by (ftime_max - ftime_min + 1)
standardization: yes # yes/no
standardization_ref_period: [1981, 2010] # if null, will use whole period
standardization_handle_infinity: no # yes/no
```

SPI:

```
return_spi: no # yes/no
Nmonths_accum: 3 # any integer covered by (ftime_max - ftime_min + 1)
standardization: yes # yes/no
standardization_ref_period: # if null, will use whole period
standardization_handle_infinity: no # yes/no
```

SUNSET: module Indicators

Threshold indicators:

- Selected Threshold
- Climate-sensitive disease indicators (CSDI):
 - Climate suitability for malaria transmission
 - Climate suitability for ticks questing activity

Indicators:

SelectedThreshold:

```
return_thresholdbased: no # yes/no
threshold: [[-2,7], [7,12], [20,Inf]] # lower and upper threshold for each requested variable
returnValues: yes # returns values or NA vs returns 1 or 0 (or NA matching original NA)
threshold_percentile: no # yes/no NOT YET DEVELOPED
```

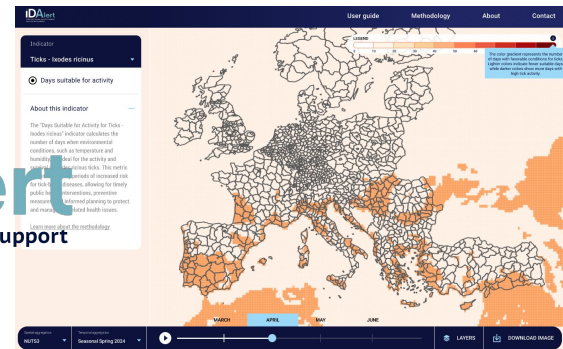
Malaria:

```
return_climate_suitability: yes # yes/no
ssp: ['P.falciparum', 'P.vivax'] # select one or several of the allowed options
```

Ticks:

```
return_climate_suitability: no # yes/no
ssp: ['I.ricinus'] # select one or several of the allowed options
```

iDAAlert
Infectious Disease decision-support
tools and Alert systems



SUNSET: module Indicators

Example recipe SPEI and Malaria indicator:

[Find full recipe in Gitlab](#)

Analysis:

```
Horizon: seasonal
Variables:
  name: tas, tdps, tasmin, tasmax, prlr
  freq: monthly_mean
  units: {tas: C, tdps: C, tasmin: C, tasmax: C, prlr: mm}
Datasets:
  System:
    - {name: ECMWF-SEAS5.1}
  Multimodel: no
  Reference:
    - {name: ERA5-Land}
Time:
  sdate: '0601'
  fcst_year: 2024
  hcst_start: '1981'
  hcst_end: '2010'
  ftime_min: 1
  ftime_max: 6
Regrid:
  method: bilinear
  type: none
```

Workflow:

```
Indicators:
  SPEI:
    return_spei: yes
    PET_method: hargreaves
    Nmonths_accum: 3
    standardization: yes
    standardization_ref_period:
    standardization_handle_infinity: yes
  Malaria:
    return_climate_suitability: yes
    ssp: ['P.falciparum', 'P.vivax']
```

SUNSET: module Indicators

Example script SPEI and Malaria indicator:

```
# Load modules
source("modules/Loading/Loading.R")
source("modules/Units/Units.R")
source("modules/Indicators/Indicators.R")

# Read recipe
recipe_file <- 'recipe.yml'
recipe <- prepare_outputs(recipe_file)

# Load datasets
data_raw <- Loading(recipe)

# Change units: very important for these indicators!
data_units <- Units(recipe, data_raw)

# Obtain SPEI and Malaria indicators according to recipe
result <- Indicators(recipe, data_units)
```

[Find full script in Gitlab](#)

SUNSET: module Indicators

Example result SPEI and Malaria indicator:

```
dim(result$SPEI$obs$data)
```

dat	var	sday	sweek	syear	time	latitude	longitude	ensemble
1	1	1	1	30	6	101	151	1

```
dim(result$SPEI$fcst$data)
```

dat	var	sday	sweek	syear	time	latitude	longitude	ensemble
1	1	1	1	1	6	10	15	51

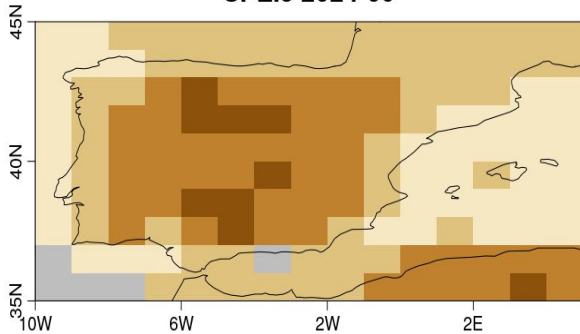
```
dim(result$Malaria$p.vivax$hcst$data)
```

dat	var	sday	sweek	syear	time	latitude	longitude	ensemble
1	1	1	1	30	6	10	15	25

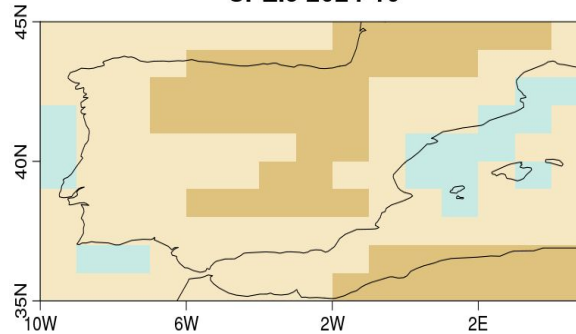
SUNSET: module Indicators

Forecast SPEI3:

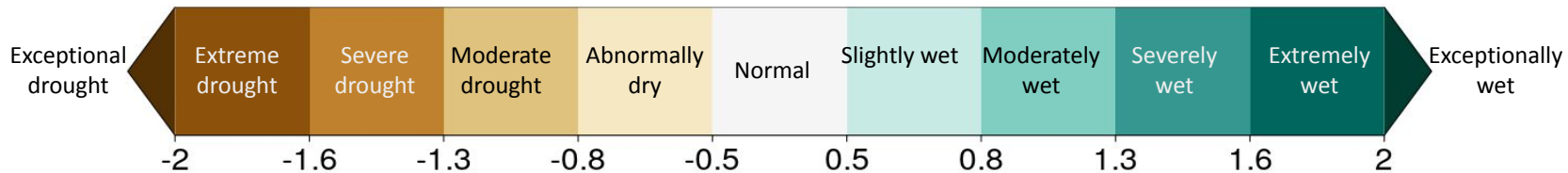
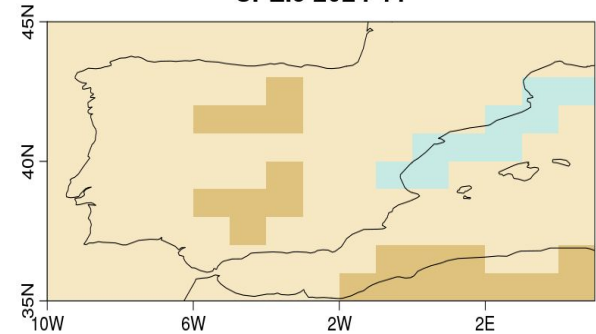
SPEI3 2024-09



SPEI3 2024-10



SPEI3 2024-11

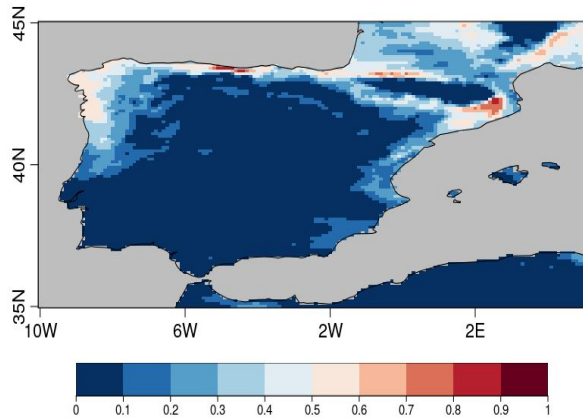


Classification adapted from Marco Turco *et al* 2017 *Environ. Res. Lett.* 12 084006 and following MeteoSwiss

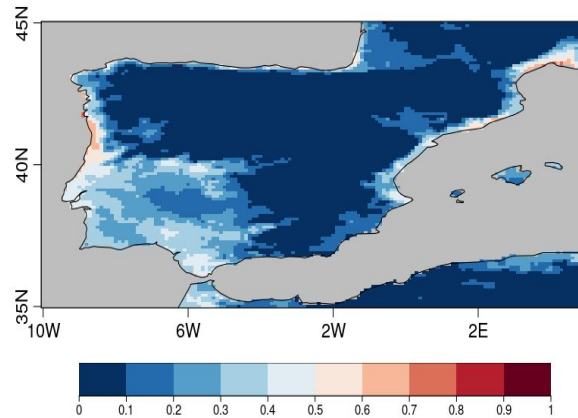
SUNSET: module Indicators

Climate suitability for malaria (*P. vivax*) 1981-2010:

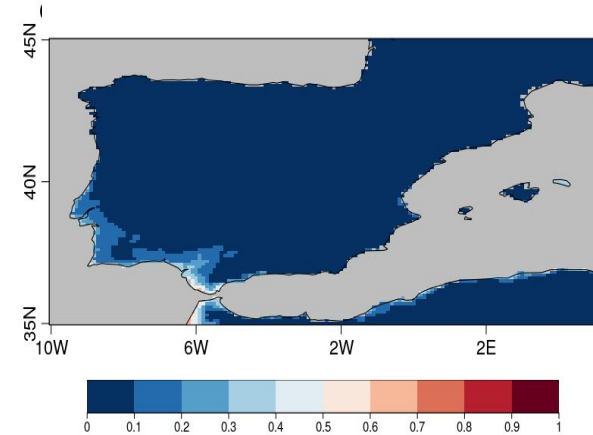
September



October



November



SUNSET: module Indicators

Status and links to documentation and examples:

- currently in branch dev-indicators
- to follow the development:
 - <https://earth.bsc.es/gitlab/es/sunset/-/issues/74>
- recipe template:
 - https://earth.bsc.es/gitlab/es/sunset/-/blob/dev-indicators/recipe_template.yml
- example script:
 - https://earth.bsc.es/gitlab/es/sunset/-/blob/dev-indicators/example_scripts/example_indicators.R
- future developments:
 - workflow to correctly calculate the indicators with full cross-validation:
 - improvement in CST_PeriodStandardization to allow for a non-consecutive period
 - improvement of the Indicators function to allow calibration before the final step
 - development of threshold_percentile of SelectedThreshold

Q&A



**Barcelona
Supercomputing
Center**

Centro Nacional de Supercomputación

Thanks for joining