Wikiprint Book

Title: Single point selection

Subject: TracMeteo - udg/ecoms/RPackage/examples/pointSelection

Version: 16

Date: 10/16/2021 01:32:00 PM

Table of Contents

Single point selection

The following call to `loadECOMS` will load a time series of surface (2m) daily mean temperature (`var = "tas"`, as defined in the ?vocabulary) for the coordinate -3.7E 40.4N, (`lonLim = -3.7`, `latLim = 40.4`) corresponding to the city of Madrid (Spain), corresponding to the summer (JJA, `season = 6:8`) of 2001 (`years = 2001`) as forecasted the previous March (`leadMonth = 2`) by the CFSv2 hindcast (`dataset = "CFSv2_seasonal_16"`). We will select the first 10 members (`members = 1:10`). Note that the original variable is stored in the CFSv2 database as 6-hourly. Hence, we indicate to the funciton to compute the daily mean from the 6-hourly data using the argument `time = "DD"`.

```
> point.cfs <- loadECOMS(dataset = "CFSv2_seasonal_16", var = "tas", members = 1:10, lonLim = -3.7, latLim = 40.4, season
[2014-06-17 15:51:15] Defining homogeneization parameters for variable "tas"
NOTE: daily mean will be calculated from the 6-h instantaneous model output
[2014-06-17 15:51:15] Defining geo-location parameters
[2014-06-17 15:51:16] Defining initialization time parameters
[2014-06-17 15:51:20] Retrieving data subset ...
[2014-06-17 15:54:16] Done
> print(object.size(point.cfs))
32088 bytes
```

The returned object contains all the necessary information for data representation (geo-location, time ...). In the next lines we plot the loaded time series for each member. The element `Data` contains the data itself. In this case, it is a 2D array with the dimensions `time` (92 days for June, July and August) and `member` (10 members selected), as indicated by the `dimensions` attribute:

```
> str(point.cfs$Data)
num [1:92, 1:10] 16.3 17.7 18.6 19.7 20.2 ...
- attr(*, "dimensions")= chr [1:2] "time" "member"
```

In the following example we plot the time series with the multi-member mean and its dispersion (interquartilic range 25-75):

```
> quartiles <- apply(point.cfs$Data, MARGIN = 1, FUN = quantile, probs = c(.25,.75))
> ens.mean <- rowMeans(point.cfs$Data)
> dates <- as.POSIXlt(point.cfs$Dates$start, tz="GMT")
> plot(dates, ens.mean, ylim = range(point.cfs$Data), ty = 'n', ylab = "tas - Daily Mean", xlab = "time")
> polygon(x = c(dates, rev(dates)), y = c(quartiles[1, ], rev(quartiles[2, ])), border = "transparent", col = rgb(0,0,1,.4
> lines(dates, ens.mean)
```

The interface `loadECOMS` is also used in the case of gridded observations. For instance, using the same geo-location and time arguments used in the previous selection, we can now retrieve the observed temperature for that particular grid cell accessing the WFDEI dataset:

```
> point.wfdei <- loadECOMS(dataset = "WFDEI", var = "tas", lonLim = -3.7, latLim = 40.4, season = 6:8, years = 2001, time
[2014-06-17 16:05:55] Defining homogeneization parameters for variable "tas"
[2014-06-17 16:06:55] Defining geo-location parameters
[2014-06-17 16:06:55] Defining time selection parameters
[2014-06-17 16:06:55] Done
> print(object.size(point.wfdei))
13360 bytes
```

It is now possible to overlay the observed temperature series with the 2-month ahead prediction of CFS for that particular grid cell (the result is not very impressive, though):

```
> lines(dates, point.wfdei$Data, col = "red", lwd = 1.5)
```