Wikiprint Book

Title: Analysing model drift in South-western Iberia

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

Version: 15

Date: 10/22/2021 03:01:39 AM

Table of Contents

Analysing model drift in South-western Iberia

In this practice, we will analyse the model drift by using the forecast of daily mean surface temperature for July 2001 considering 6 different forecast (lead) months, rfom January to June. The lead month 0 (i.e., the initialization of July itself) will be the reference from which we will compute the anomalies. In this example, we will consider the first member of the CFSv2 hindcast.

```
ref <- loadECOMS(dataset = "CFSv2_seasonal_16", var = "tas", members = 1, lonLim = c(-10,-1), latLim = c(36,40), season =
```

Messages on-screen inform about the loading process. Note that the selection of `leadMonth = 0` will give a NOTE on screen:

```
[2014-09-03 17:50:20] Defining homogeneization parameters for variable "tas"
NOTE: daily mean will be calculated from the 6-h model output
NOTE: 'leadMonth = 0' selected
[2014-09-03 17:50:20] Defining geo-location parameters
[2014-09-03 17:50:20] Defining initialization time parameters
[2014-09-03 17:50:26] Retrieving data subset ...
[2014-09-03 17:50:31] Done
```

```
plotMeanField(ref)
title(main = "Lead month 0 forecast of July 2001")
# This is the spatial mean of the reference field
ref.field <- apply(ref$Data, MARGIN = c(3,2), FUN = mean, na.rm = TRUE)
```

Next, we will load the forecast of the target variable recursively for lead month values from 1 to 6 (i.e., the initializations from January to June). The different objects are arranged in a list:

```
cfs.list <- lapply(1:6, function(lead.month) {
     loadECOMS(dataset = "CFSv2_seasonal_16", var = "tas", members = 1, lonLim = c(-10,-1), latLim = c(36,40), season = 7,
})
```

In order to visualize the departures of each lead month from the reference in the same range of values, we will use the `spplot` method for plotting spatial objects of the library `sp`. To this aim, we will first compute the multi-member spatial mean for each lead month forecast, and then we will arrange the data in a matrix of 6 columns (one for each month), and x * y rows, as follows:

```
# The library sp needs to be installed to do this example:
require(sp)
# Matrix of anomalies between lead month and reference
aux.mat <- sapply(1:length(cfs.list), function(i) {apply(cfs.list[[i]]$Data, MARGIN = c(3,2), FUN = mean, na.rm = TRUE) -
# 2D coordinates
xy <- expand.grid(ref$xyCoords$x, ref$xyCoords$y)
# This step ensures regularity of the CFS grid, which is not perfectly regular:
xy.coords <- coordinates(points2grid(points = SpatialPoints(xy), tolerance = .003))
# Now we create a data.frame with the coordinates X-Y in the first two columns and the mean anomalies in the next 6 column
df <- cbind.data.frame(xy.coords, aux.mat)
names(df) <- c("x","y",paste("LeadMonth_",1:6, sep = ""))
str(df)
'data.frame':   55 obs. of  8 variables:
$ x          : num  -10.31 -9.38 -8.44 -7.5 -6.56 ...
$ y          : num  40.2 40.2 40.2 40.2 40.2 ...
$ LeadMonth_1: num  0.0596 0.1601 0.5955 0.9068 1.2601 ...
$ LeadMonth_2: num  -0.1215 0.0509 0.3622 0.5444 0.6977 ...
$ LeadMonth_3: num  -0.48 -0.359 -0.402 -0.693 -0.967 ...
$ LeadMonth_4: num  0.22303 0.27295 0.23869 0.04764 -0.00855 ...
$ LeadMonth_5: num  -1.212 -0.986 -0.682 -0.587 -0.584 ...
$ LeadMonth_6: num  -1.314 -0.732 -0.32 -0.5 -0.98 ...
coordinates(df) <- c(1,2)
gridded(df) <- TRUE
```

```
class(df)
```

Which returns the new spatial object class:

```
[1] "SpatialPixelsDataFrame"
attr(,"package")
[1] "sp"
```

In the next lines we use apply the `spplot` method of package `sp`, generating a lattice-type map. In first place, we will also load a `SpatialLines` dataset remotely stored at Santander Met Group server, in order to represent the coastline in the lattice map generated as a geographical reference:

```
load(url("http://meteo.unican.es/work/downscaler/aux/wlines.rda"), verbose = TRUE)
l1 <- list("sp.lines", wlines)
spplot(df, as.table = TRUE, col.regions = colorRampPalette(c("blue","white","red")), at = seq(-5.25,5.25,.25), scales = li
```

The results show how a increasing lead month leads to a negative bias of the forecast, demonstrating that the mean state of a variable of a forecast is not stationary along the runtime dimension.

Finally, we display the spatial mean of the anomalies w.r.t. the reference for each lead month considered using a barplot:

```
barplot(colMeans(df@data), names.arg = abbreviate(names(df)), xlab = "lead month", ylab = "anomaly (ºC)")
title(main = "Mean forecast bias w.r.t. the lead-month 0 initialization")
mtext("Member 1")
```