Visualizing geographic predictions of genetic offset measures

Olivier François

Introduction

Genomic offset statistics predict the maladaptation of populations to rapid habitat alteration based on association of genotypes with environmental variation. This brief tutorial explains how to represent predictions of genomic offset statistics within geographic maps. This can be achieved using standard `R` packages dedicated to spatial analysis.

In the tutorial, spatial prediction of genomic offset will be illustrated by analyzing publicly available genomic data from 1,096 European lines of the model plant *Arabidopsis thaliana* (https://www.1001genomes.org/) and climate models described in the sixth IPCC report (IPCC AR6).

To represent geographic maps, there are many other and often better methods than the ones presented in the tutorial. I do not claim being a cartography specialist and the approach below is likely suboptimal. It is at least quite flexible, easy to reproduce, and can be used as basis for improved representations.

Loading genomic and environmental data

Displaying genomic offset statistics in geographic space will require that the `R` packages `terra`, `geodata`, `fields`, and `maps` are installed. The tutorial will use `LEA` for performing a genotype-environment association study and for computing genomic offset statistics.

```r
# Required packages
# Loading worldclim/cimp6 bioclimatic data
library(terra)
```

```
## terra 1.7.71
```

```r
library(geodata)

# displaying images and maps
library(fields)
```

```
## Loading required package: spam
```

```
## Spam version 2.10-0 (2023-10-23) is loaded.
## Type 'help( Spam)' or 'demo( spam)' for a short introduction
## and overview of this package.
## Help for individual functions is also obtained by adding the
## suffix '.spam' to the function name, e.g. 'help( chol.spam)'.
```

```
##
## Attaching package: 'spam'
```

```
## The following objects are masked from 'package:base':
##
##     backsolve, forwardsolve
```

```
## Loading required package: viridisLite
```

```
##
## Try help(fields) to get started.

##
## Attaching package: 'fields'

## The following object is masked from 'package:geodata':
##
##     world

## The following object is masked from 'package:terra':
##
##     describe
```

```r
library(maps)

# Adjusting genotype-environment association models
library(LEA)
```

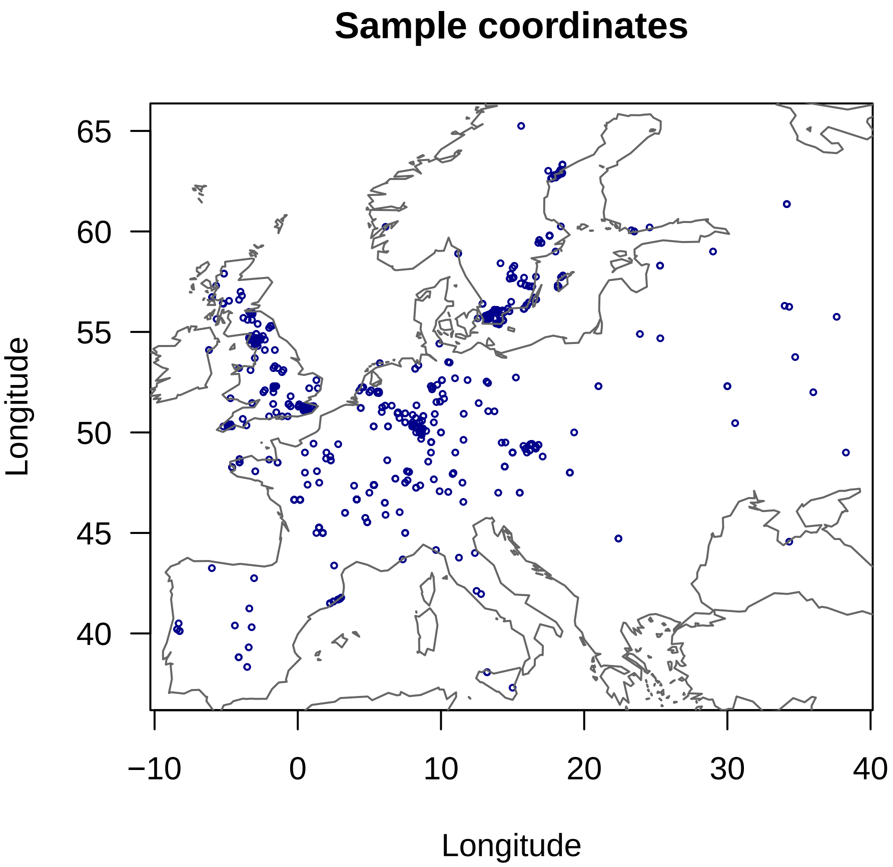
Genomic and geographic data for A. thaliana samples are available from a previous tutorial on **running structure-like population genetic analyses with R**. The data contain 1,096 genotypes from the first chromosome of the plant and geographic coordinates (latitude and longitude) associated with each sample. They can be downloaded as follows.

```r
# default timeout option is 60s -- increase to 300s
options(timeout = max(300, getOption("timeout")))

# download sample genotypes in the working directory (54.4 MB)
url = "http://membres-timc.imag.fr/Olivier.Francois/Arabidopsis/A_thaliana_chr1.geno"
download.file(url = url, destfile = "./A_thaliana_chr1.geno")

# download sample coordinates in the working directory
url = "http://membres-timc.imag.fr/Olivier.Francois/Arabidopsis/at_coord.coord"
download.file(url = url, destfile = "./at_coord.coord")
```

The data are then loaded as R objects, and the sample coordinates can be visualized as follows.

```r
genotype = LEA::read.geno("./A_thaliana_chr1.geno")
coordinates = as.matrix(read.table("./at_coord.coord"))
```

```r
plot(coordinates, cex = .4, col = "darkblue",
     xlab = "Longitude", ylab = "Longitude",
     main = "Sample coordinates", las = 1)

maps::map(add = TRUE, interior = FALSE, col = "grey40")
```



Bioclimatic variables

In the following presentation, predictions will be based on bioclimatic variables from the worldclim database. The bioclimatic variables are downloaded below. The `climate` object contains 19 historical temperature and precipitation variables with low resolution. A temporary path is used for downloading. If additional analyses are planed, changing to non-temporary folders could be useful to avoid reloading (which may be slow).

```r
# Download global bioclimatic data from worldclim
climate <- geodata::worldclim_global(var = 'bio',
                                     res = 10,
                                     download = TRUE,
                                     path=tempdir())
```

Next, the `climate_future` object contains future temperature and precipitation variables predicted from the SSP2-4.5 scenario developed with respect to the sixth IPCC report (IPCC AR6). There are several prediction models, and the one used here is called 'ACCESS-ESM1-5'.

```r
# Download future climate scenario from 'ACCESS-ESM1-5' climate model.
climate_future <- geodata::cmip6_world(model='ACCESS-ESM1-5',
                                       ssp='245',
                                       time='2041-2060',
                                       var='bioc',
                                       download = TRUE,
                                       res=10,
                                       path=tempdir())
```

Now, environmental data can be extracted for each sample site. The extraction command results in an environmental matrix `X.env` having 1,096 rows and 19 columns after removing IDs.

```
# extracting historical environmental data for A. thaliana samples
X.env = terra::extract(x = climate,
                       y = data.frame(coordinates),
                       cells = FALSE)
# remove IDs
X.env = X.env[,-1]

# extracting future environmental data for A. thaliana samples
#X.env_fut = terra::extract(x = climate_future, y = data.frame(coordinates), cells=FALSE)
#X.env_fut = X.env_fut[,-1]
```

Genotype-Environment Association study

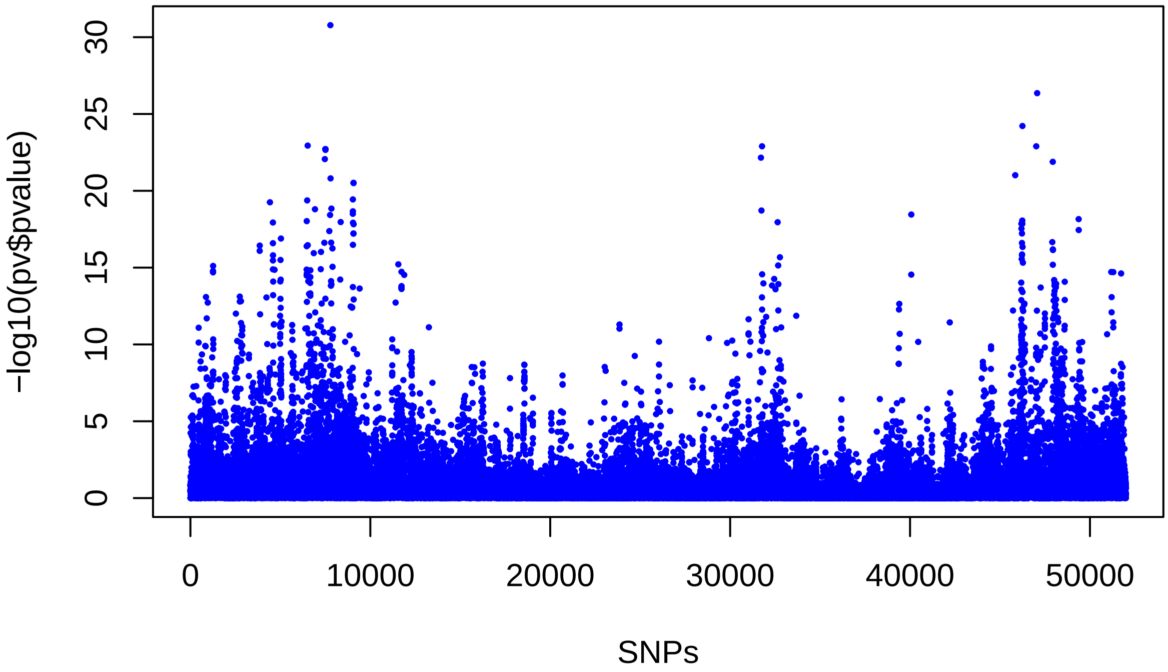
To evaluate genomic offset statistics, environmental effect sizes must be estimated at each genomic locus. This can be achieved by applying a latent factor model in `LEA`. Based on a previous analysis, five latent factors are used. Because temperature and precipitation have distinct units, the environmental data will be scaled to unit standard deviation. In other words, the data will be centered by substracting their mean, and then divided by their standard deviation.

Another approach reduces the dimension of the bioclimatic data set by performing separate scaled PCA on temperature and precipitation variables. New variables could then be defined by retaining the first components in each separate analysis. This is not implemented here because our sample size is large.

```
# latent factor GEA model
mod_lfmm = LEA::lfmm2(input = genotype,
                      env = scale(X.env),
                      K = 5,
                      effect.sizes = TRUE)

# get environmental effect sizes
B <- mod_lfmm@B
```

The GEA model can also be used to define a subset of candidate loci to be included in genomic offset computation.

```
pv = lfmm2.test(mod_lfmm, input = genotype, env = scale(X.env), full = TRUE)
plot(-log10(pv$pvalue),
     xlab = "SNPs",
     cex = .3, pch = 19, col = "blue")
```



A set of candidate loci needs to be relatively large. Statistical significance is not a requirement here, and a cut-off threshold at 5 can be chosen (but try 0 to 4).

```
# define candidate loci for GO analysis
candidates =  -log10(pv$pvalue) > 5

# taking all loci for GO analysis
# candidates =  -log10(pv$pvalue) > 0

# how many candidate loci?
sum(candidates)
```

```
## [1] 1328
```

Extracting historical and future climate for Europe

First, a reasonable range of longitude and latitude coordinates must be defined. This range of values includes the European mainland and some islands. Below `nc` is a critical resolution parameter. Increasing its value produces more precise maps at higher costs.

```
## nc = resolution, higher is better but slower
nc = 200

# range of longitude for Europe (deg E)
long.mat <- seq(-10, 40, length = nc)

# range of latitude for Europe (deg N)
lat.mat <- seq(36, 67, length = nc)

# matrix of cells for Europe (nc times nc)
coord.mat <- NULL
for (x in long.mat)
  for (y in lat.mat) coord.mat <- rbind(coord.mat, c(x,y))
```

Then, the `R` package `terra` can extract the climate and the future climate data for every cell defined in the above coordinate matrix.

```
# Extract historical climate
env.new = terra::extract(x = climate,
                         y = data.frame(coord.mat),
                         cells = FALSE)
env.new = env.new[,-1]

# Extract future climate
env.pred = terra::extract(x = climate_future,
                          y = data.frame(coord.mat),
                          cells=FALSE)
env.pred = env.pred[,-1]
```

Computing genomic offset for environmental matrices

The R package LEA can compute genomic offset statistics by using the genomic.offset function, but this function does not allow missing environmental data (NA's). For terrestrial species, environmental NA's are, however, a (tricky) way to display data in land areas only, showing seas as blank areas.

To use that trick, genomic offset statistics will be recalculated without the help of the genomic.offset function, which, by the way, is not complicated. As the lfmm was adjusted on scaled historical environmental predictors, the same scaling must be performed for the future data. This is done below.

```
## scaling bioclimatic variables (with the same scale as in the lfmm)
m.x <- apply(X.env, 2, FUN = function(x) mean(x, na.rm = TRUE))
sd.x <- apply(X.env, 2, function(x) sd(x, na.rm = TRUE) )

env.new <- t(t(env.new) - m.x) %*% diag(1/sd.x)
env.pred <- t(t(env.pred) - m.x) %*% diag(1/sd.x)
```

For example, consider a (particular example) site in Germany with longitude around 10.06689 E, and latitude around 50.30769.

```
# Coordinates (long, lat) of a geographic site in Germany, Europe
coord.mat[36139,]
```

```
## [1] 35.22613 57.49749
```

The genomic offset at this particular geographic location can be calculated as follows. The statistic corresponds to the geometric GO defined in (Gain et al. 2023).

```
## Geometric genomic offset for coord.mat[36139,]
mean(((env.new - env.pred)[36139,]  %*% t(B[candidates,]))^2)
```

```
## [1] 0.08187216
```

Displaying a map requires to repeat the above computation for all geographic locations in the coordinate matrix. In R, this is usually be achieved by using the apply function. Below, a less elegant approach is carried out to evaluate the genomic offset at each cell. The reason for using a slow method is that the faster method may overload the memory space. So, be patient or modify the code chunk to avoid the loop.

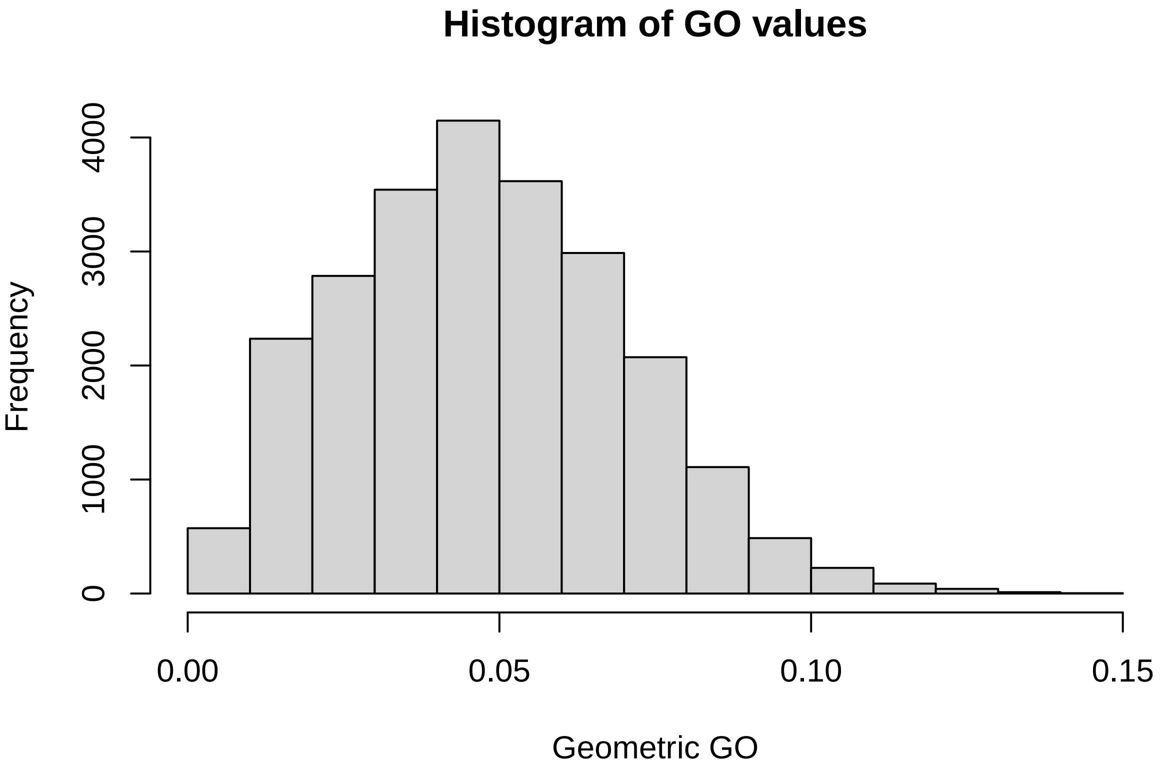
```
## gg contains the Gain et al. geometric GO computed at each matrix cell
## be patient, it may be very slow for large nc
gg = NULL
for (i in 1:nrow(env.new)){
gg[i] = mean(((env.new - env.pred)[i,]  %*% t(B[candidates,]))^2, na.rm = TRUE)
}
```

The matrix that corresponds to the mapping of genomic offset statistics can be obtained as follows.

```
## matrix of genomic offset for the Europe map
## NA when below sea level.
go = t(matrix(gg, byrow =  FALSE, ncol = nc))
```

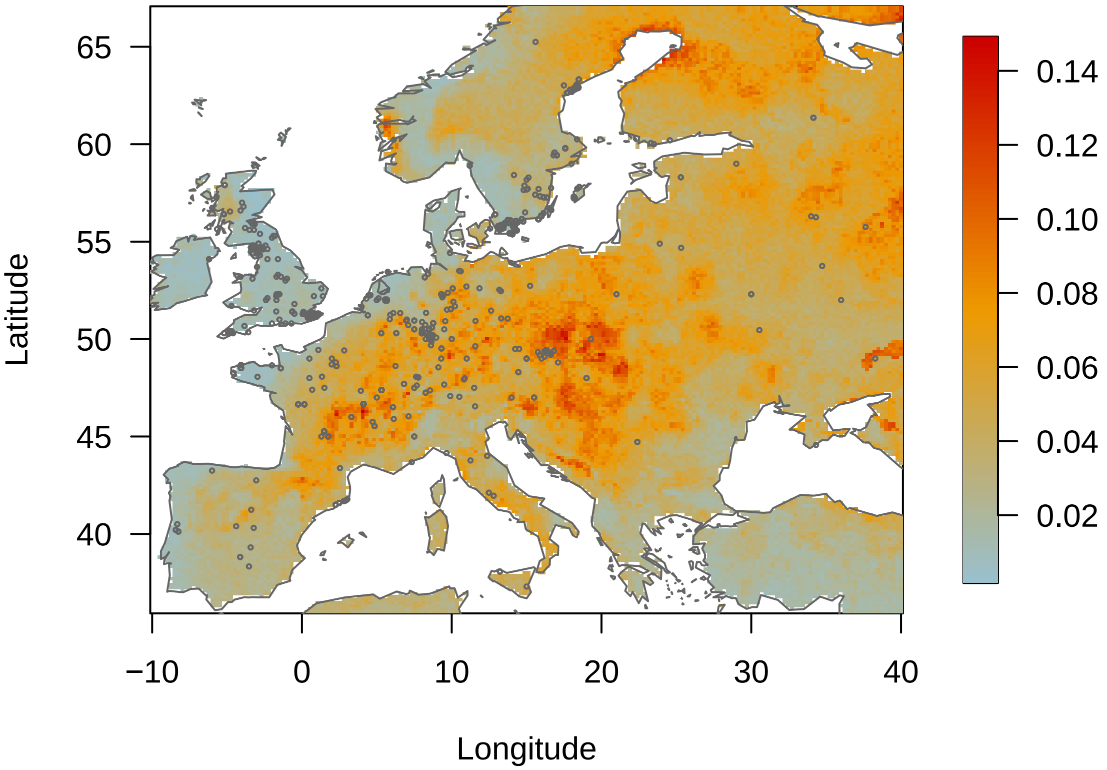
Let us check the histogram of GO statistics.

```
hist(as.numeric(go),
     main = "Histogram of GO values",
     xlab = "Geometric GO")
```



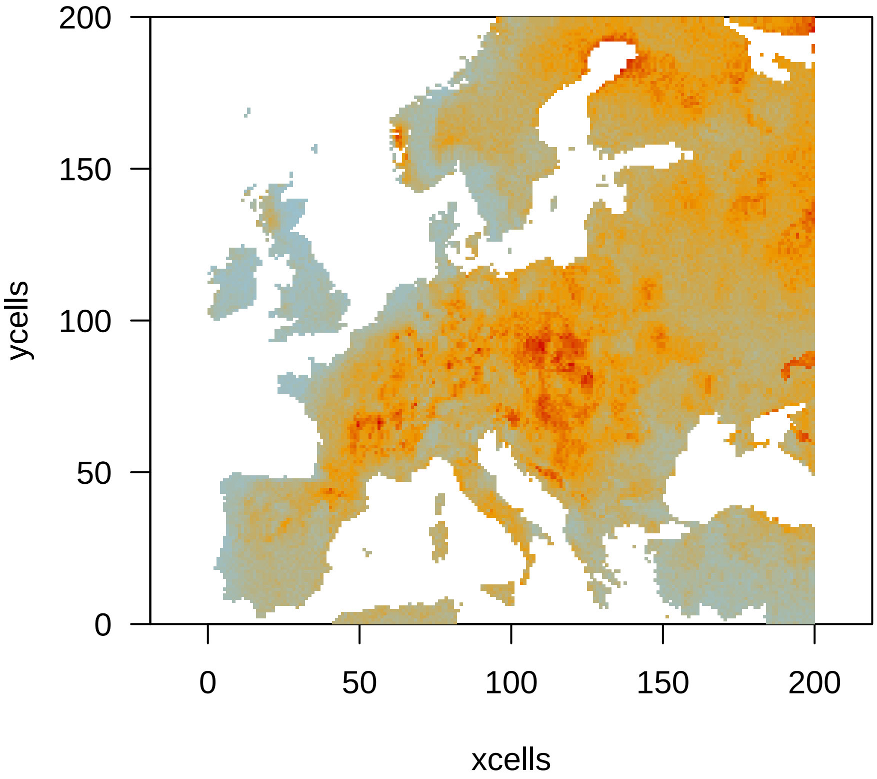
There are several ways to represent the GO matrix in `R`. One option is by using the `R` package `fields`. I like this option because there is a legend placed to the right of the figure.

```
# my colors - they might change the story!
my.colors = colorRampPalette(c("lightblue3", "orange2", "red3"))(100)

## bins extreme values above .1 - see histogram
# go2  = go
# go2[go2 > .1] = .1

fields::image.plot(long.mat, lat.mat, go,
                   col = my.colors,
                   las = 1,
                   xlab = "Longitude",
                   ylab = "Latitude")

## add contour of Europe and sample locations
maps::map(add = TRUE, interior = FALSE, col = "grey40")
points(coordinates, col = "grey40", cex = .3)
```



Another graphic option is to use `image` in the R package `terra` after conversion as raster. This might then be harder to read the figure axes as longitude and latitude. But the resulting map looks the same as above.

```
r <- terra::rast(t(go)[nc:1,])
terra::image(r,
             col = my.colors,
             las = 1,
             xlab = "xcells",
             ylab = "ycells")
```



For A. thaliana, the most pessimistic predictions of maladaptation under scenario SSP2-4.5 are for areas in France, Italy, Belgium, Germany and in Central Europe. Regions in the Alps, Northern Europe and in areas under oceanic influence appear to be at lower risk. Of course, this interpretation is specific to the bioclimatic variables considered con and the IPPC scenario used. The result requires further evidence from additional scenarios and combinations of predictors.