

# Fitting ENM Generalized Linear Model

Emilio Berti

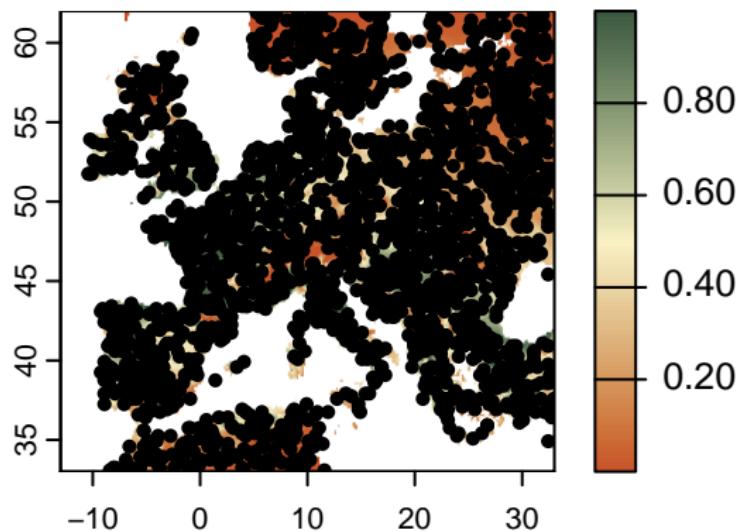
## Virtual species niche

```
library(terra)
set.seed(123)

niche <- function(x, mu, sigl, sigr) {
  x <- x[!is.na(x)]
  out <- rep(NA, length(x))
  out[x < mu] <- exp(- ((x[x < mu] - mu) / sqrt(2) / sigl)
  out[x >= mu] <- exp(- ((x[x >= mu] - mu) / sqrt(2) / sigr)
  return(out)
}

# actual niche
mu <- c(12, 800)
sigl <- c(3, 300)
sigr <- c(4, 500)
```

# Virtual species occurrence data



## Virtual species environmental data

A data ready for ENM/SDM usually looks like this:

	BI001	BI012	occ
225	11.333906	767	1
1279	5.691469	640	0
1585	9.607145	1285	0
946	13.927315	759	1
1578	6.552542	619	0

## Fitting GLM

In R, `glm(<formula>, <data>, family = binomial("logit")`) fits a GLM to binary (0/1) data.

logit is the inverse function of the logistic equation:  $\log\left(\frac{p}{1-p}\right)$ , where  $p$  is a probability.

```
m <- glm(  
  occ ~ poly(BI001, 2, raw = TRUE) + poly(BI012, 2, raw = TRUE),  
  data = d,  
  family = binomial("logit"))
```

In R `I(x^n)` means *as is*, i.e. it does not square the value, it evaluates the polynomial.

## Inspecting fitted GLM

```
summary(m)$coefficients
```

		Estimate	Std. Error	z
(Intercept)		-1.694289e+01	8.615173e-01	-19.66633
poly(BI001, 2, raw = TRUE)1		1.804445e+00	1.108938e-01	16.27184
poly(BI001, 2, raw = TRUE)2		-6.983801e-02	4.786003e-03	-14.59214
poly(BI012, 2, raw = TRUE)1		1.723085e-02	1.402416e-03	12.28655
poly(BI012, 2, raw = TRUE)2		-9.748549e-06	8.234337e-07	-11.83890

## Niche inferred by GLM

```
x <- seq(4, 22, length.out = 50)
y <- seq(0, 2000, length.out = 50)
z <- matrix(NA, nrow = length(y), ncol = length(x))
z_inferred <- z

for (j in seq_along(x)) {
  z[j, ] <- niche(x[j], mu[1], sigl[1], sigr[1]) *
    niche(y, mu[2], sigl[2], sigr[2])
  z_inferred[j, ] <- predict(
    m,
    newdata = data.frame(BI001 = x[j], BI012 = y),
    type = "response"
  )
}
```

## Niche inferred by GLM

```
image(  
  x, y, z,  
  xlab = "BIO01",  
  ylab = "BIO12",  
  col = hcl.colors(100, "Fall", rev = TRUE)  
)  
contour(  
  x, y, z,  
  add = TRUE,  
  levels = seq(0.1, 1, by = 0.2),  
  lwd = 2  
)  
contour(  
  x, y, z_inferred,  
  add = TRUE,  
  levels = seq(0.1, 1, by = 0.2),  
  col = "white", lwd = 2  
)
```

## Niche inferred by GLM

Symmetrical, but not that bad in this case.

