

# Análisis de presencias con procesos de puntos

Tutorial intermedio de spatstat

---

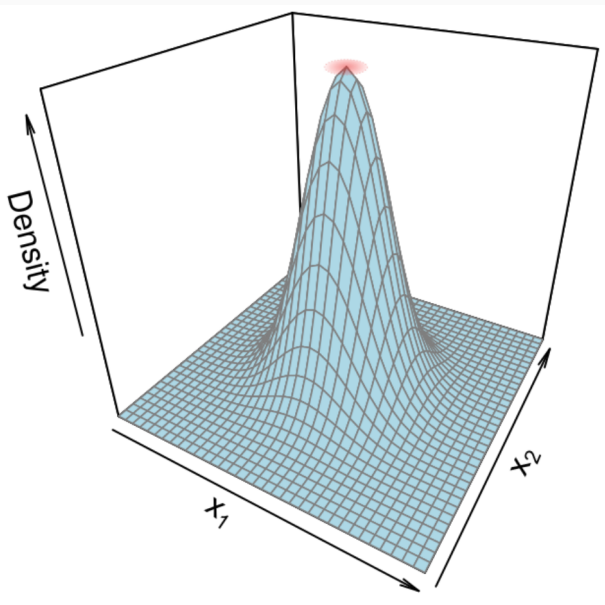
Gerardo Martín

2022-06-29

## Simulación de presencias

---

## Especificación de un centroide



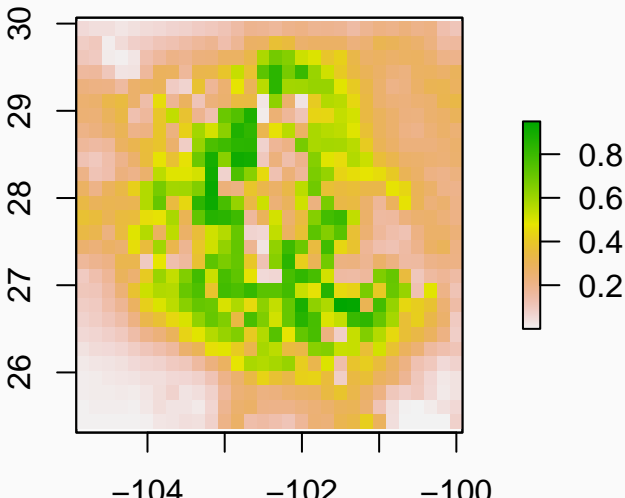
## Código - generando favorabilidad “verdadera”

```
centroide <- cellStats(r, mean)
r.df <- data.frame(rasterToPoints(r))
covar <- cov(r.df[, 3:5])
md <- mahalanobis(r.df[, 3:5], center = centroide, cov = covar)
head(md)

## [1] 5.846738 6.383437 6.443874 7.296541 6.475630 6.066614
```

## Código - viendo la favorabilidad

```
md.r <- rasterFromXYZ(data.frame(r.df[, 1:2], md))  
md.exp <- exp(-0.5*md.r)  
plot(md.exp)
```

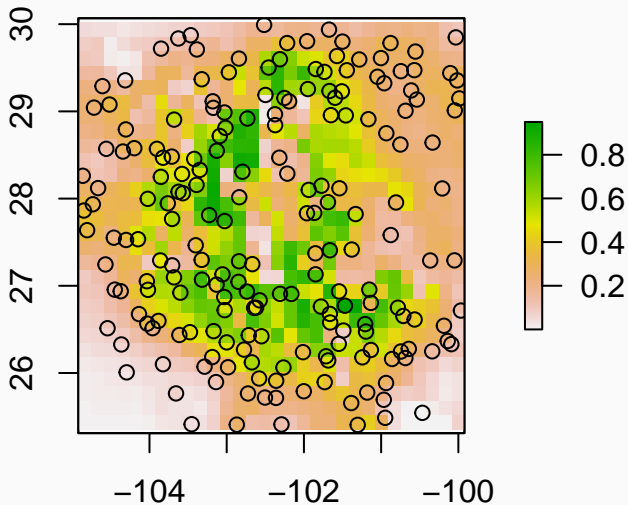


```
set.seed(182)
puntos.2 <- dismo::randomPoints(mask = md.exp,
                                n = 200,
                                prob = T)

## Warning in .couldBeLonLat(x, warnings = warnings): CRS is NA.
## longitude/latitude

puntos.2 <- data.frame(puntos.2)
puntos.2$x <- puntos.2$x + rnorm(200, 0, 0.05)
puntos.2$y <- puntos.2$y + rnorm(200, 0, 0.05)
```

```
plot(md.exp); points(puntos.2)
```



## Formateo para spatstat

---



```
source("Funciones-spatstat/imFromStack.R")  
source("Funciones-spatstat/winFromRaster.R")  
source("Funciones-spatstat/plotQuantIntens.R")
```

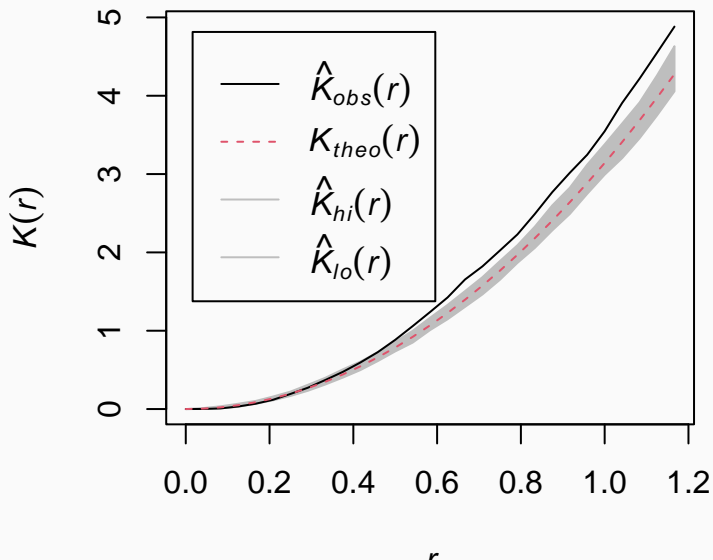
```
r.im <- imFromStack(r)
w <- winFromRaster(r)
puntos.2.ppp <- ppp(x = puntos.2$x,
                    y = puntos.2$y,
                    window = w,
                    check = F)
Q <- pixelquad(X = puntos.2.ppp, W = as.owin(w))
```

## Análisis exploratorio

---

```
K <- envelope(puntos.2.ppp, fun = Kest, nsim = 39)

## Generating 39 simulations of CSR ...
## 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 1
##
## Done.
```

**K**

1. Pareciera que el proceso está levemente autocorrelacionado
2. No sabemos de momento si afectará al modelo
3. Debemos poner atención al modelo ajustado

```
plotQuantIntens(imList = r.im,  
                noCuts = 5,  
                Quad = Q,  
                p.pp = puntos.2.ppp,  
                dir = "",  
                name = "Respuestas-centroide")
```

```
## pdf
```

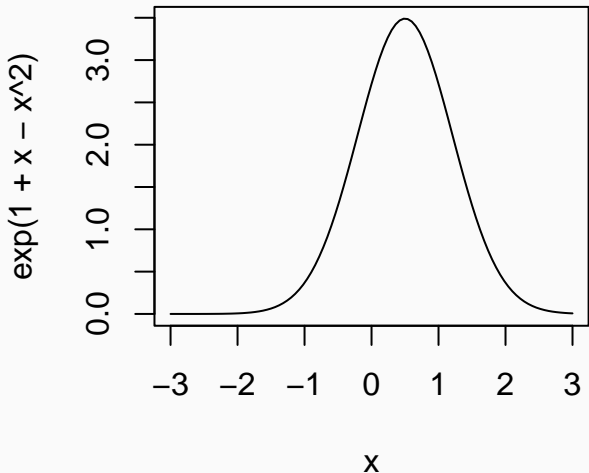
```
## 2
```

Ver archivo de gráficas

## Consideraciones para proponer modelos

Curvas con forma de campana → fórmula cuadrática

```
curve(exp(1 + x - x^2), from = -3, 3)
```





Ecuación lineal:

$$y = \alpha + \beta_1 x_1 + \cdots + \beta_n x_n$$

Ecuación polinomial de 2º grado

$$y = \alpha + \beta_1 x_1 + \beta'_1 x_1^2 + \cdots + \beta_n x_n + \beta'_n x_n^2$$

Recordemos que  $y = \log \lambda$

## ¿Qué variables podemos incluir en el mismo modelo?

**Regla de oro:** Aquellas que no estén correlacionadas

- Que  $x_1$  no sea predictor de  $x_2$
- No se puede atribuir efecto de  $x_1$  ó  $x_2$  sobre  $\lambda$
- Necesitamos medir correlación entre pares de variables (**pairs**)

# Medición de correlación entre covariables

`pairs(r)`

