GADMTools - Graphics

jean.pierre.decorps@gmail.com

2021-08-04

# Graphics

functions

Function	Description
choropleth	Draws a choropleth on selected regions
classDots	Plots dots on a map with values between different fixed classes
dots	Plots dots on a map
dotDensity	Draws a dot-density map
isopleth	Draws an isopleth on selected regions
gadm_showNorth	Displays a north arrow on a plot
gadm_showScale	Displays a scale on a plot

Display a north arrow on a plot

```r
library(GADMTools)
data("Corsica")

plotmap(Corsica) %>% gadm_showNorth("tl")
```

Display a scale on a plot

```
library(GADMTools)
data("Corsica")

plotmap(Corsica) %>% gadm_showNorth("tl") %>% gadm_showScale("bl")
```
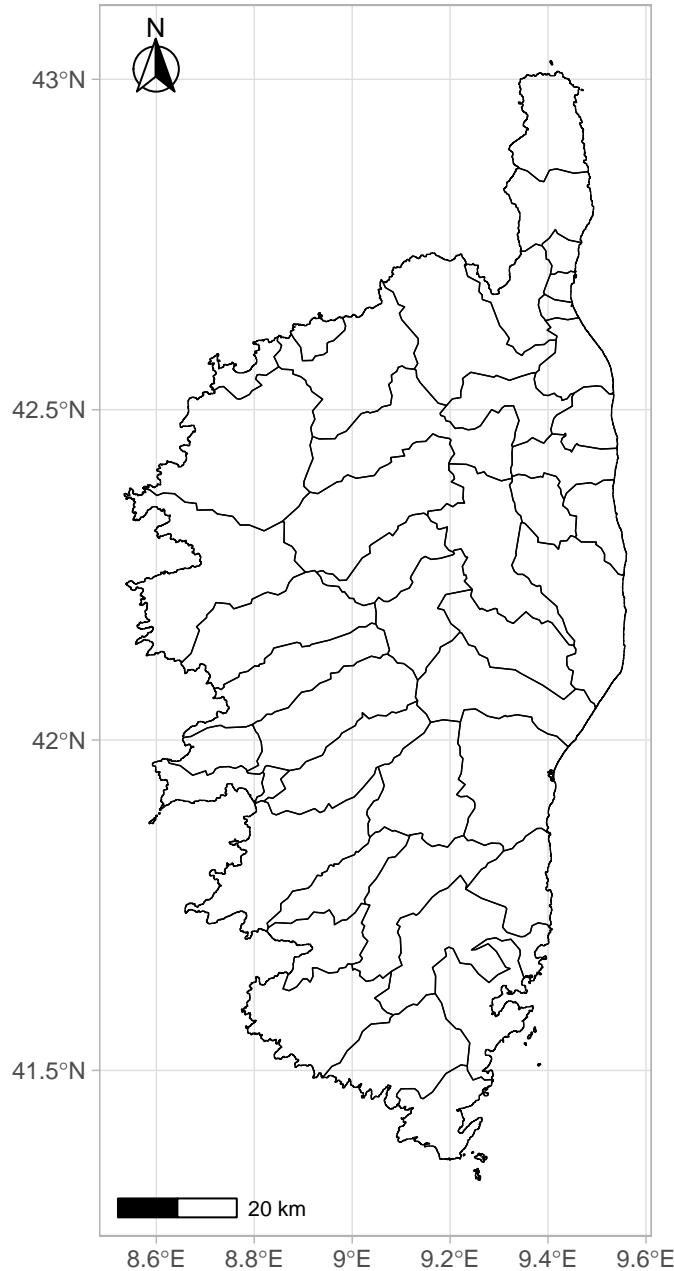
Plotting dots on a map

```r
# Preparing data.frame
# --------------------
data("Corsica")

Corse <- gadm_union(Corsica, 0)
Cantons <- listNames(Corsica, 4)
L <- length(Cantons)
Pop <- floor(runif(L, min=15200, max=23500))

longitude <- runif(6, min=8.74, max = 9.25)
latitude  <- runif(6, min=41.7, max = 42.6)
Cases <- floor(runif(6, 25, 80))

Type <- rep(c("TYPE A", "TYPE B", "TYPE C"), 6, length.out = 6)
DAT <- data.frame(longitude, latitude, Cases)
```

```
# Simple dots
#--------------------------------------------------------------------
dots(Corsica, DAT, color="red", size = 3)
```

```
# Classified dots
#------------------------------------------------------------------
dots(Corse, points = DAT,
     palette = "Reds",
     value="Cases")
```

```r
# Typed points
#--------------------------------------------------------------------
DAT2 <- data.frame(longitude, latitude, Type)
dots(Corse, points = DAT2,
     color = "#ee00ee",
     strate="Type")
```

Plotting proportionals dots

```r
# Test of propDots with default parameters
# ------------------------------------------------------------------------------
propDots(Corse,
         data = DAT,
         value="Cases",
         color = "blue")
```

```r
# Test of propDots with defined breaks
# ------------------------------------------------------------------------------
propDots(Corse,
         data = DAT,
         value="Cases", breaks=c(30, 40, 50, 70, 100),
         color = "blue")
```

```
# Test of propDots with forced range of breaks
# ----------------------------------------------------------------------------
propDots(Corse, data = DAT, value="Cases",
         breaks=c(0, 25, 50, 75, 100),
         range = c(25, 100))
```

Plotting dots with classified size

```r
library(GADMTools)

classDots(Corse, DAT, color="blue", value = "Cases", steps = 4)
```

Dot-Density

```
library(GADMTools)
data("Corsica")

# Creates test data.frame -------------------------------------------------
# -------------------------------------------------------------------------
VAR_1 <- as.integer(runif(n = 43, min = 800, max = 15800))
VAR_2 <- as.integer(runif(n = 43, min = 1000, max = 15800))
VAR_3 <- as.integer(runif(n = 43, min = 1500, max = 15800))
Cantons <- listNames(Corsica, 4)
DF <- data.frame(Cantons, VAR_1, VAR_2, VAR_3, stringsAsFactors = FALSE)

dotDensity(Corsica,
                DF, adm.join="Cantons", dot.size = 0.5, cases.by.dots = 1000,
                values = c("VAR_1", "VAR_2", "VAR_3"),
                labels = c("H1N1", "H1N2", "H2N2"),
                palette = c("#ffff00", "#ffaa00", "#FF3200"))
```

Plotting density

```
library(GADMTools)

isopleth(Corse, data = DAT, palette = "Blues")
```

Plotting a choropleth

```r
DAT <- data.frame(Cantons, Pop, stringsAsFactors = FALSE)
choropleth(Corsica, data = DAT, value = "Pop", adm.join = "Cantons",
           breaks = "sd", palette = "Greens")
```

fast.choropleth()

fast.choropleth(

```
x, data, value=NULL,

breaks = NULL, steps = 5,

adm.join=NULL, legend = NULL,

labels = NULL,

palette=NULL, title=""

)
```

Parameter	Description
x	**Object** GADMWrapper
data	**data.frame** - data to plot
value	**String** - the name of the column in the data.frame we want to plot (eg: an incidence in epidemiology studies)
breaks	
steps	**Integer** - number of breaks. Default = 5. If breaks is NOT NULL this value is used internally with cut().
adm.join	**String** - the name in GADM spdf dataset which will be joined with a column of the data.
legend	**String** - legend title. Default NULL.
labels	**String vector** labels for the legend. Default NULL
palette	**String** - An RColorBrewer palette name or a String vector vector of colors. Default NULL.
title	**String** - Title of the plot. Default is an empty string.

Example

```r
MAP <- gadm_sp_loadCountries("BEL", level = 3, simplify=0.01)
DAT = read.csv2("BE_clamydia_incidence.csv")

# Rewriting District names
# ------------------------
DAT$district <- as.character(DAT$district)
DAT[7,1] = "Brussel"
DAT[20,1] <- "Liège"
DAT[22,1] = "Marche-en-Famenne"
DAT[27,1] = "Neufchâteau"
DAT <- rename(DAT, NAME_3 = district)


fast.choropleth(MAP, DAT,
                adm.join = "NAME_3",
                value = "rate03",
                steps = 4,
                breaks = "jenks",
                palette="Greens",
                legend = "Incidence",
                title="Chlamydia incidence by Belgian district (2003)")
```
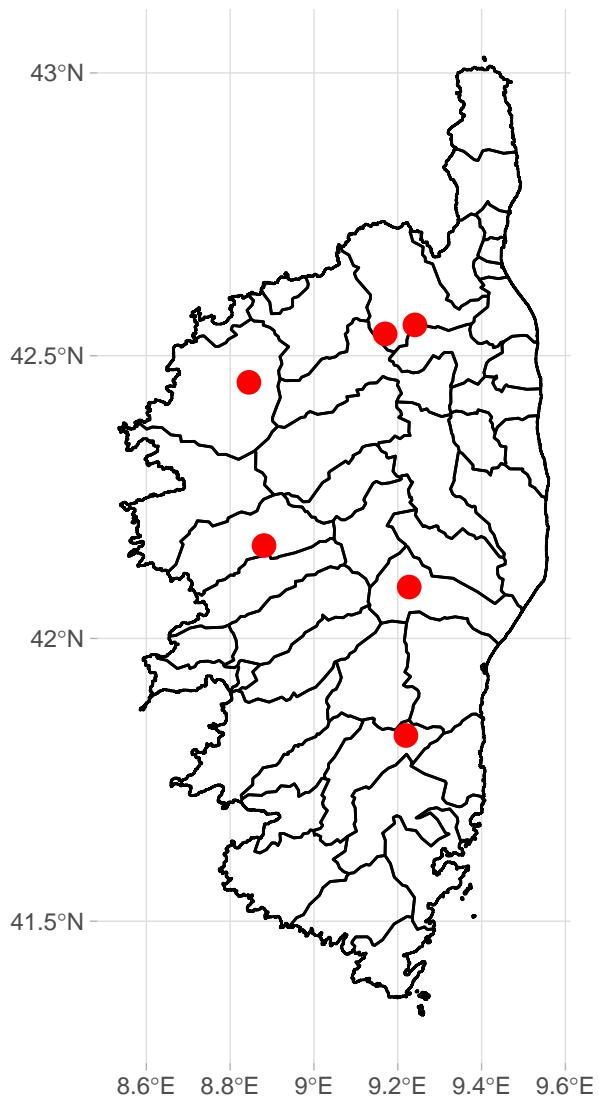
Figure 1: Simple dots
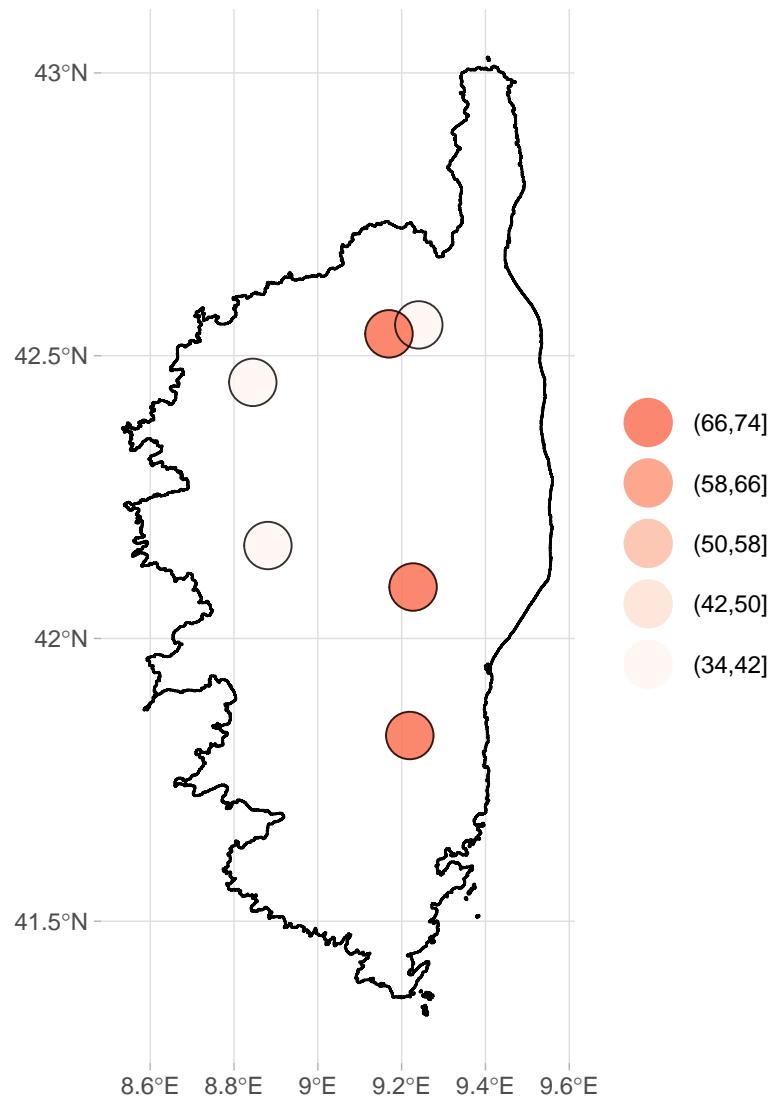
Figure 2: Classified dots
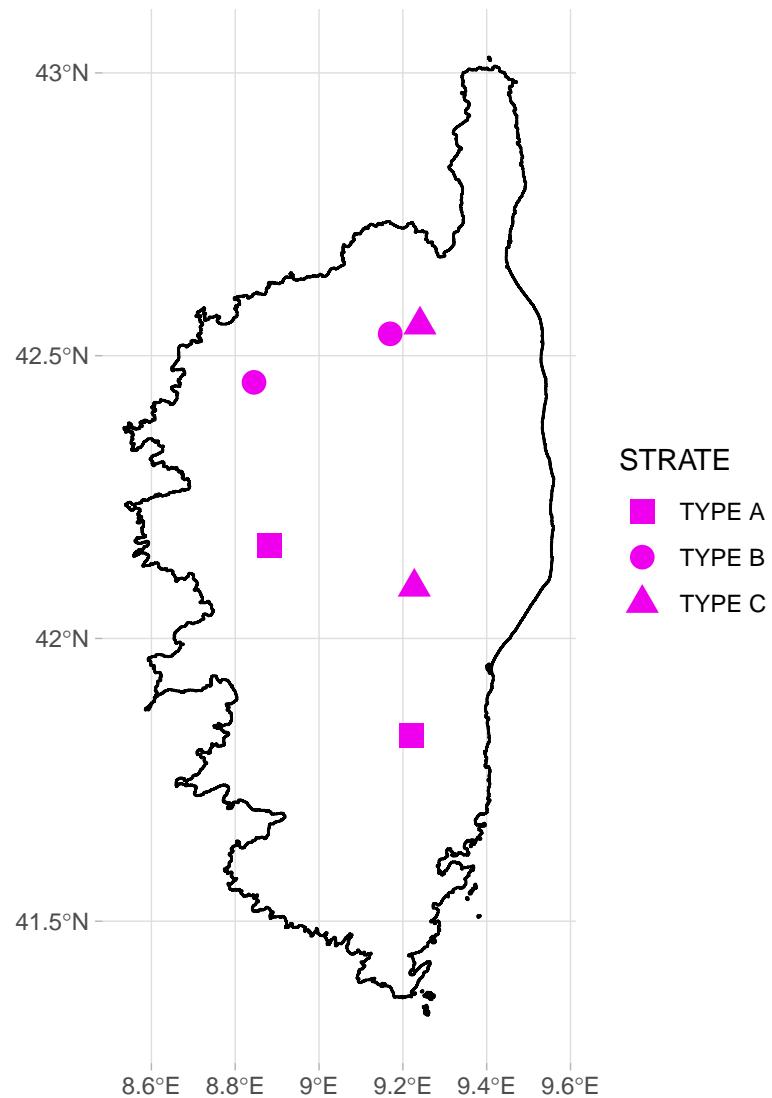
Figure 3: Typed dots
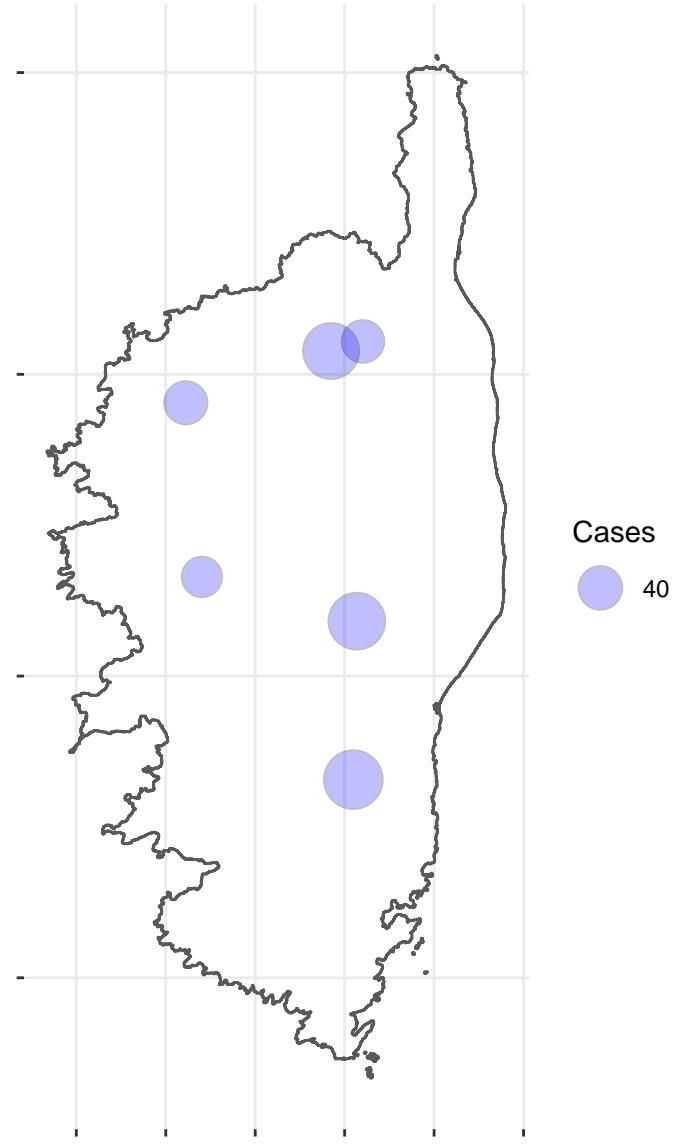
Figure 4: propDots
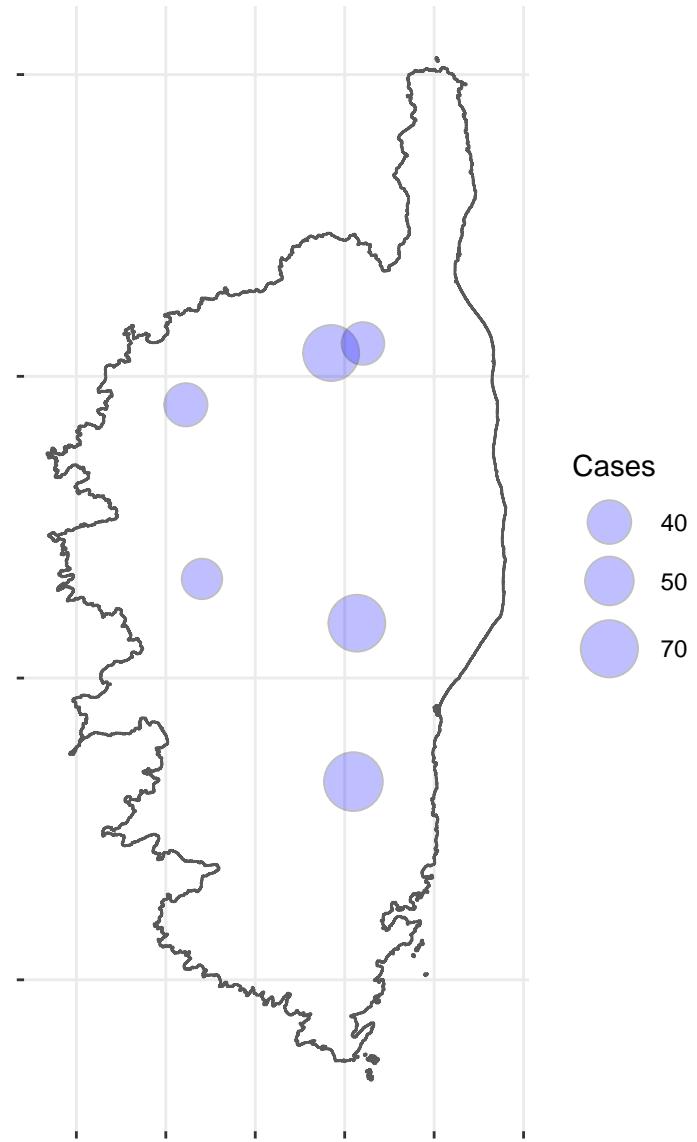
Figure 5: propDots with provided breaks
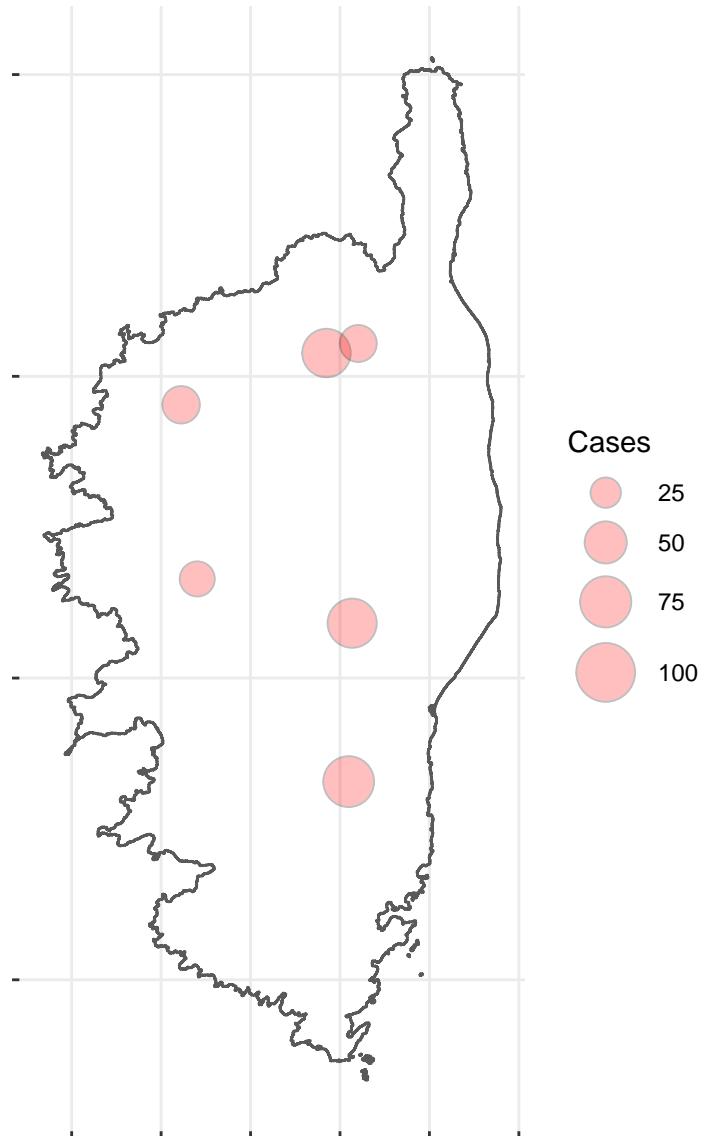
Figure 6: propDots
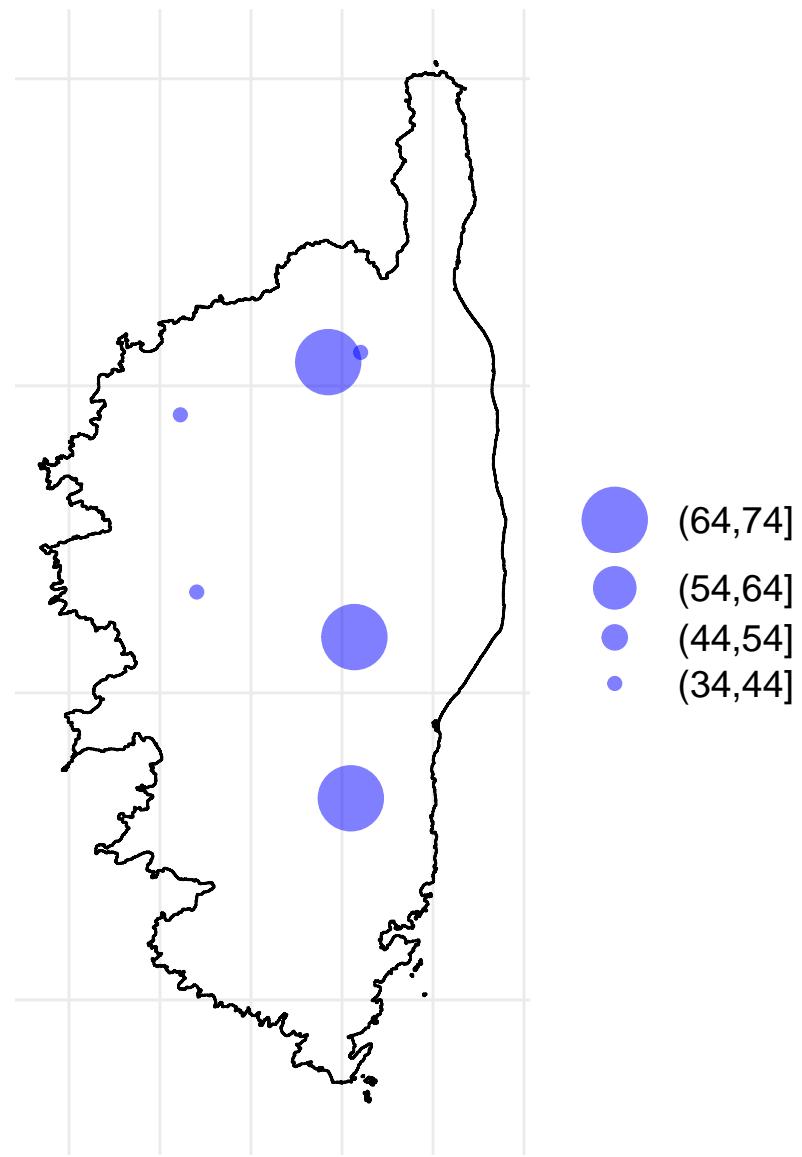
Figure 7: classDots
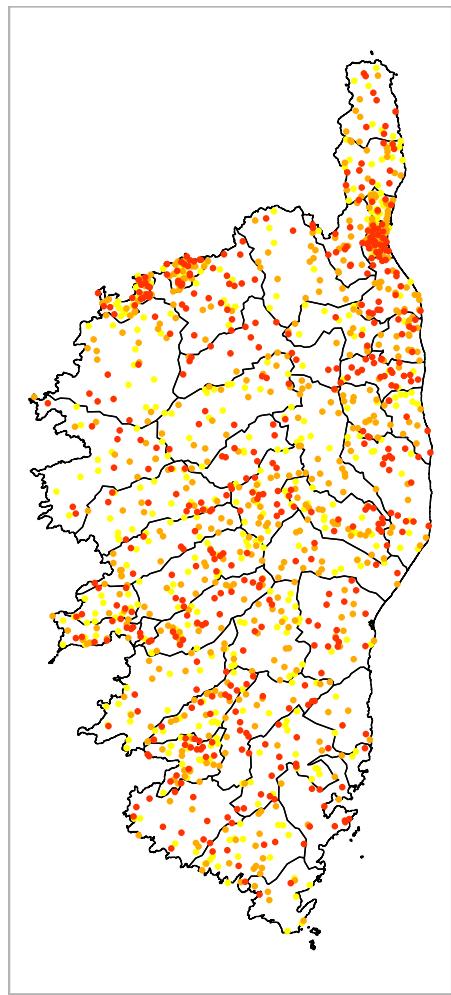
1 dot = 837 cases

- • H1N1
- • H1N2
- • H2N2

Figure 8: Dot-Density
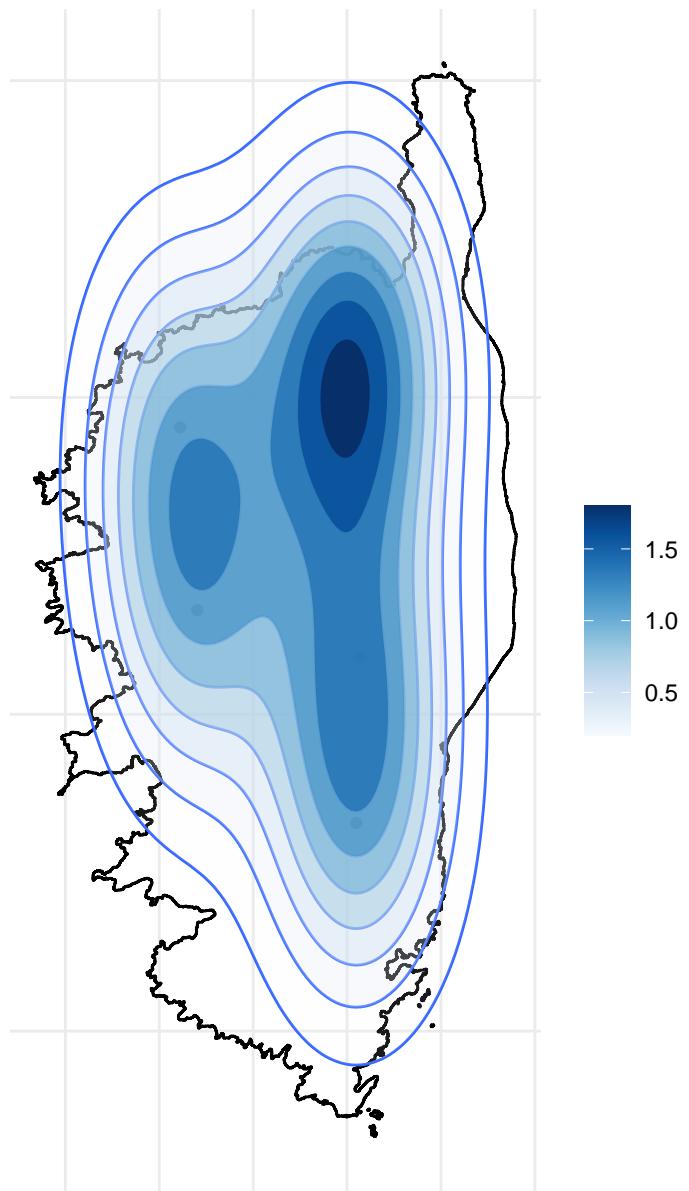
Figure 9: Isopleth

Figure 10: Choropleth

Figure 11: drawing a fast.choropleth