Analysis of continuous spatialized data

Dr. Vladimir Grosbois
vladimir.grosbois@cirad.fr

CIRAD
UR AGIRs
Type of data

• Data on the value of a variable (usually a quantitative variable) measured at different locations

• Moreover that variable
  • can have a value at every possible spatial location
  • Rainfall over the month of October 2018
  • or can have a value only at certain spatial locations
    • The diameter of trees of a particular species

• However the data always consist in measurements of the variable at discrete (discontinuous) spatial locations
Exemple of continuous spatialized data

• Data on the proportion of trees in plots of a wildlife park in Tchad damaged by elephants
• Two spatial objects

• placette.data is a SpatialPointDataFrame that contains
  • the position of the placettes
  • an attribute reflecting the tree species in the plot
  • an attribute reflecting the proportion of trees damaged

• parc.sp is SpatialPixelDataFrame that
  • divides the park in pixels
  • each pixel has as an attribute the dominant tree species
summary(placette.data)
Object of class SpatialPointsDataFrame
Coordinates:

<table>
<thead>
<tr>
<th>min</th>
<th>max</th>
</tr>
</thead>
<tbody>
<tr>
<td>coords.x1</td>
<td>19.36569 19.99069</td>
</tr>
<tr>
<td>coords.x2</td>
<td>10.57639 11.04764</td>
</tr>
</tbody>
</table>
Is projected: FALSE
proj4string :
[+proj=longlat +ellps=WGS84 +towgs84=0,0,0,0,0,0,0 +no_defs]
Number of points: 329
Data attributes:

<table>
<thead>
<tr>
<th>PLOT</th>
<th>DAMAGE</th>
<th>zone</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Min. :0.00000</td>
<td>Acacia  :114</td>
</tr>
<tr>
<td>2</td>
<td>1st Qu.:0.03571</td>
<td>Combretaceae:215</td>
</tr>
<tr>
<td>3</td>
<td>Median :0.16667</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Mean :0.30800</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>3rd Qu.:0.50000</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Max. :1.00000</td>
<td></td>
</tr>
</tbody>
</table>
spplot(placette.data)
Exemple of continuous spatialized data

`summary(parc.sp)`

Object of class `SpatialPixelsDataFrame`

Coordinates:

```
   min   max
x 19.35056 19.99856
y 10.56806 11.07806
```

Is projected: `FALSE`

`proj4string`:

```
[+proj=longlat +ellps=WGS84 +towgs84=0,0,0,0,0,0,0 +no_defs]
```

Number of points: 26006

Grid attributes:

```
   cellcentre.offset cellsize cells.dim
x 19.35206   0.003  216
y 10.56956   0.003  170
```

Data attributes:

```
   zone
   Acacia   : 6353
   Combretaceae:19653
```
spplot(parc.sp)
Spatial dependency in spatialized data

- **Spatial dependency**: the values of the variable measured at spatially close points are more similar than the values of the same variable measured at spatially remote points
  - Stationnarity: when spatial dependency is similar at all points in a study area.
  - Non stationnarity: when spatial dependency varies within a study area.

- **Isotropy**: when the intensity of spatial dependency is not affected by the direction.
- **Anisotropy**: when the intensity of spatial dependency is affected by the direction.
Statistics used to assess spatial dependency

- Variogram: representation of the variance within pairs of values of variable as a function of distance.

- The Variogram can be modelled.
The variogram function plots the variance as a function of distance computed from the data.

```r
library(gstat)
damage.var0 <- variogram(DAMAGE~1, placette.data, boundaries=seq(1, 40, 1))
```

- A model formula
- The SpatialDataFrame object including the data
- The limits of the distance categories for which variance will be computed
The variogram function plots the variance as a function of distance computed from the data.

\[ \text{plot(damage.var0)} \]
Variogram for elephant damages

• The variogram function can produce variograms that account for the effect of covariates.
• The covariate effects are declared in the model formula

```r
damage.var1 <- variogram(DAMAGE ~ zone + I(x^2) + I(y^2) + I(x*y) + x + y, placette.data, boundaries = seq(1, 40, 1))
```

In this formula, we account for the effects of zone (type of vegetation and of a second degree polynomial effect of the coordinates)
Variogram for elephant damages

plot(damage.var1)
Variogram characterization and modelling

![Variogram diagram with labels: range, sill, partial sill, nugget.](image-url)
The function `vgm` generates a variogram model.
For doing so, we have to declare the characteristics of the variogram:

```r
vgmod0 <- vgm(psill = 0.03, model = "Exp", range = 10, nugget = 0.03)
```

```r
plot(damage.var1, vgm0)
```
Variogram modelling

- The function `fit.variogram` fits a variogram model to the data variogram model.
- It requires a baseline variogram model (here we use `vgm0`) which will be updated to fit as well as possible the observed variogram.

```r
vgmod1 <- fit.variogram(damage.var1, vgm0)
plot(damage.var1, vgm1)
```
To obtain a continuous spatial representation of a variable from discrete data

Example:
• A network of meteorological station records rainfall in a region
• We wish to obtain a continuous representation of rainfall in that region
A method for spatial interpolation: spatial kriging

- It produces predictions of the value of the variable at locations where it has not been measured. The prediction is a weighted average of the values of the variable at the locations where it has been measured.

- The weights are functions of the distance between the location where we wish to predict the value of the variable and the locations where the variable has been measured.

- Effects of covariates can also be accounted for to derive the predictions.

- It is necessary to have previously characterized the spatial dependency structure: semi-variogram.
Spatial kriging for the elephant damages

We use the krig function

degmoy.kr<-krige
  (DAMAGE~zone+I(x^2)+I(y^2)+I(x*y)+x+y,
   placette.data,parc.sp,
   vgmmod1)

In the krig function,
• the formula account for covariate effects
• a variogram model is required to characterize spatial dependency (vgmod1)
spplot(degmoy.kr[1])  spplot(placette.data,"DAMAGE")