Non-stationary geostatistics using R/gstlearn

Thomas Romary

Option Géostatistique 2024

Contents

This practical requires all questions from the last two practicals (on variography and kriging) to be answered. The database is the same, and focus is on the logarithm of the cadimum.

1 Preamble

```
rm(list=ls()) #Clean the working directory

library(minigst)
```

2 Preamble

3 Preamble

```
OptCst_defineByKey("ASP",0)
```

3.1 Loading the Meuse dataset

```
library(sp) # load library
data(meuse) # load data
data(meuse.grid) # load target grid
names(meuse.grid)[5]="dist.m" # change the name to concur with meuse
```

Create the data Db

```
## Create Db from dataframe
data=dfToDb(meuse,coordnames = c("x","y"))

## Add log concentrations
var_name = c( "cadmium", "copper", "lead", "zinc")
addVarToDb(data,var=log(meuse[,var_name]),vname = paste0("log_",var_name))

# Plot the data

dbplot_point(data,size="log_cadmium",sizeRange = c(0.1,3), pointColor = "red",
             xlab = "Easting (m)", ylab = "Northing (m)", title = "Log Cadmium")
```

Create the target (grid) Db

```
target = dfToDbGrid(meuse.grid,coordnames = c('x','y'))

# Plot the distance covariate (see universal kriging)

dbplot_grid(target, color='dist.m',title='Distance to the river')
```

4 Universal Kriging

4.1 Variography and non-stationnarity

To work with universal kriging or kriging with external drift, it suffices to call `minikriging` (to compute predictions) or `xvalid` (to perform cross-validation) with drift functions. The variogram model should be fitted on a experimental variogram computed with the residuals where the trend has been filtered out.

1. Compute an experimental variogram without then with drift `addDrift()`, compare them. What do you see ?

```
# experimental variogram on raw data
varioexp_aniso = vario_exp(data,vname="log_cadmium",dir=c(0,45,90,135),nlag=20, dlag=100.)

# plot of the experimental variogram
p1 = plot_vario(varioexp_aniso, pairDisplay = "size", title="Experimental variogram on raw data")

# experimental variogram on residuals
varioexp_aniso_drift = vario_exp(data,vname="log_cadmium",drift = 'dist.m',dir=c(0,45,90,135),nlag=20, d
p2 = plot_vario(varioexp_aniso_drift, pairDisplay = "size", title="Experimental variogram on residuals")

ggarrange(p1, p2, ncol = 2, nrow = 1,legend = "bottom",common.legend = TRUE)
```

2. Fit a model to the residuals (discard the anisotropy)

```
varioexp_drift = vario_exp(data,vname="log_cadmium",drift = 'dist.m',nlag=20, dlag=100.)

# fitting the model
struct_names=c("NUGGET","EXPONENTIAL")
```

```
model_drift=model_fit(varioexp_drift,drift = 'dist.m',struct=struct_names)
# display of the model
model_drift$display()

# plot of the experimental variogram and the fitted model
plot_vario(varioexp_drift,model = model_drift,pairDisplay = "size",
           title="Model adjustment for cadmium residuals")
```

4.2 Universal Kriging

3. Compute and plot the universal kriging (UK) using the variogram fitted on the residuals.

```
target = minikriging(data, target, vname = "log_cadmium", drift ='dist.m', model = model_drift, type =

## Plot kriging estimate
dbplot_grid(target, color='UK.log_cadmium.estim',title='Predicted log cadmium concentration')

## Plot kriging stdev
dbplot_grid(target, color='UK.log_cadmium.stdev',title='log cadmium prediction standard deviation')
```

4.3 Cross-validation

4.4 Comparison and interpretation

4. Compare ordinary kriging and universal kriging with `correlation()`. Interprete.

4.5 Adding other explanatory variables

We will try here to add *soil* and *ffreq*. Since those are categorical variables encoded as factors in the dataframe, we need to build the corresponding covariate vectors (1 less than the levels to avoid colinearity). To do this, let's consider the following piece of code:

NB: It is necessary to recreate the grid because the selection causes a bug when performed prior to the creation of the indicators.

5. Compute the variogram with the new universality conditions, fit a model and perform the kriging. Do the results improve ?

5 Maximum Likelihood estimation

Here we compute the maximum likelihood estimates of the following model for the log cadmium concentrations:

$$Cd(x) = \mu + Y(x)$$

where Cd represents the log cadmium concentration, μ a constant mean and Y a centered Gaussian random field.

We will also consider models of the form:

$$Cd(x) = \mu(x) + Y(x)$$

where $\mu(x)$ may vary according to the location, e.g. $\mu(x) = X(x)\beta$ with X an explanatory variable.

To do this, we use the **geoR** package. You may need to install it.

geoR uses a particular class to store spatial data sets : **geodata** (similar to a gstlearn db). The first thing is hence to transform our data in that class.

```
library(geoR)
meuse.geoR = as.geodata(meuse, coords.col = 1:2, data.col = 3,covar.col = 8)
meuse.geoR.grid = as.geodata(meuse.grid, coords.col = 1:2,covar.col = 5)
plot(meuse.geoR)
```

We can also compute variograms in **geoR**.

Omnidirectional

```
par(mfrow=c(1,2))
vg <- variog(meuse.geoR, uvec = seq(0,2000,l=20), bin.cloud = T)
plot(vg)
```

Check for anisotropies

```
vg.72 <- variog(meuse.geoR,uvec = seq(0,2000,l=20), direction = 72/180 * pi ,bin.cloud = T)
plot(vg.72)
```

Then according to the covariance structure we want to infer (exponential by default), we can try something like this (see **?likfit** for more details)

```
mean(meuse$cadmium)
var(meuse$cadmium)
ml = likfit(meuse.geoR, ini = c(0.5, 400), nug = 0.2)
summary(ml)
ml.dist = likfit(meuse.geoR, ini = c(0.5, 400), nug = 0.2,trend = trend.spatial(trend = ~dist ,meuse.ge
summary(ml.dist)
```

Likelihood ratio test

```
T1.2 = 2 *(ml$loglik-ml.dist$loglik)
1-pchisq(T1.2,1)
```

We can represent the resulting fitting on the variogram

```
plot(vg)
lines(ml)
lines(ml.dist)

plot(vg.72)
lines(ml)
lines(ml.dist)
par(mfrow=c(1,1))
```

5.1 Prediction

```
add.variable =function(var, data)
{
  data = cbind(data,var)
  gridtemp = Db()
  for (field in names(data))
    gridtemp[field] = data[field]
  gridtemp
  gridtemp$setLocators(c("x","y"), ELoc_X())

  nx = c(length(unique(data[,1])),length(unique(data[,2])))
  dx =  c(min(diff(sort(unique(data[,1])))),min(diff(sort(unique(data[,2])))))
```

```
  x0 = c(min(data[,1]),min(data[,2]))

  gridgs = DbGrid_create(nx,dx,x0)
  migrateMulti(gridtemp,gridgs,names = tail(names(data),1))
  gridgs$addSelection(!is.na(gridgs["dist"]),name="sel")

  #gridtemp.$addColumns(gridtemp.,var)
  gridgs
}
```

We perform kriging with the **geoR** functions, on the grid and on the validation locations.

```
k.grid = krige.conv(meuse.geoR,loc = meuse.geoR.grid$coords,krige = krige.control(obj.m=ml.dist,trend.d
```

Next we plot the resulting kriging and standard deviation maps using gstlearn and compute the MSE on the validation locations.

```
gridgs = add.variable(k.grid$predict, meuse.grid)
p = ggDefaultGeographic()
p = p + plot.grid(gridgs,
                  show.legend.raster = TRUE, palette="Spectral",legend.name.raster="log(ppm)")
#p = p + plot.point(dat,flagCst = T,pch=18,cex=1.5)
p = p + plot.decoration(title="Prediction by Kriging (ML based)")
ggPrint(p)


gridgs = add.variable(sqrt(k.grid$krige.var), meuse.grid)
p = ggDefaultGeographic()
p = p + plot.grid(gridgs,
                  show.legend.raster = TRUE, palette="Spectral",legend.name.raster="log(ppm)")
#p = p + plot.point(dat,flagCst = T,pch=18,cex=1.5)
p = p + plot.decoration(title="Kriging standard deviation (ML based)")
ggPrint(p)
```

6. Fit a model involving the *soil* and *freq* variables (as indicators). Perform a likelihood ratio test to ascertain their relevance.
7. Try using a Matern covariance model

6 Comparison and interpretation

8. Compare the results obtained through variogram fitting and with `correlation()`. Interprete.
9. Divide your dataset into two subsets: one for training, another for testing, possibly several times into a loop. Explain how you choose to build these two subsets and why.
10. Interpolate `cadmium` on the entire target grid from the training data, using the various methods studied this week (OK, COK, UK, etc).
11. Compare them on the test data. Which one would you pick and why ?