Geostatistics

Homework 4a

Giorgos Raptakis
374030

Contents

Task 1

From the previous exercise we choose the specific values [ID, X, Y, chemical elements] of 1773 samples of our area.

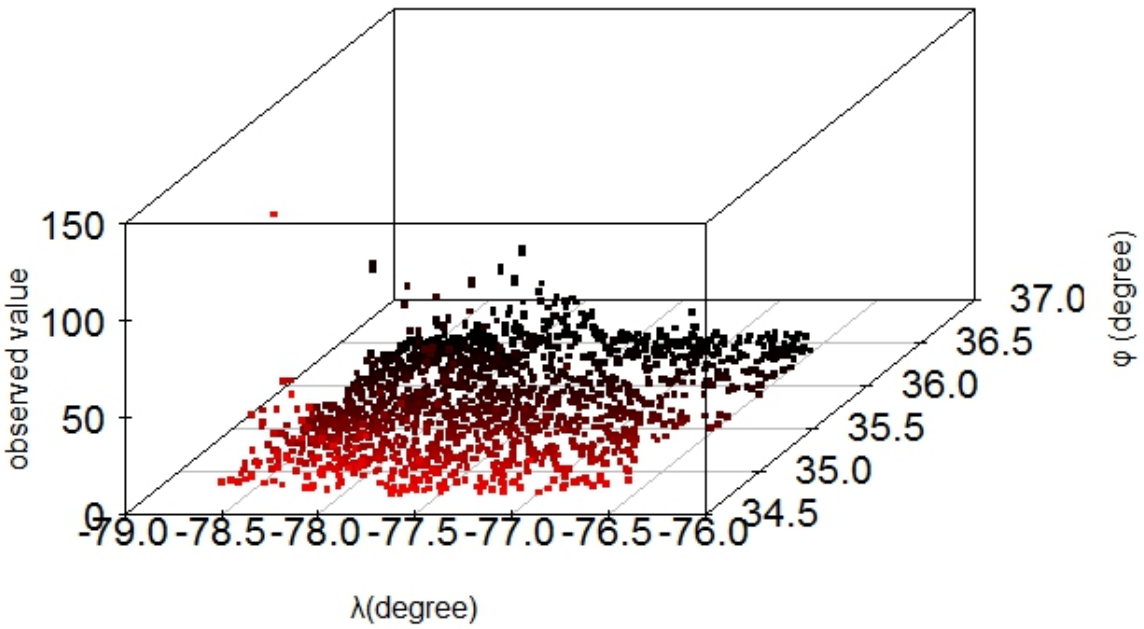
FID	U_DN_PPM	POINT_X	POINT_Y
352	1,4	-77,7279	34,6942
353	4,1	-78,1468	34,7009
355	1,5	-77,687	34,7032
358	145,6	-78,3409	34,7058
361	1,8	-77,7365	34,7073
362	7,1	-78,6112	34,7074
363	18	-77,0187	34,7074

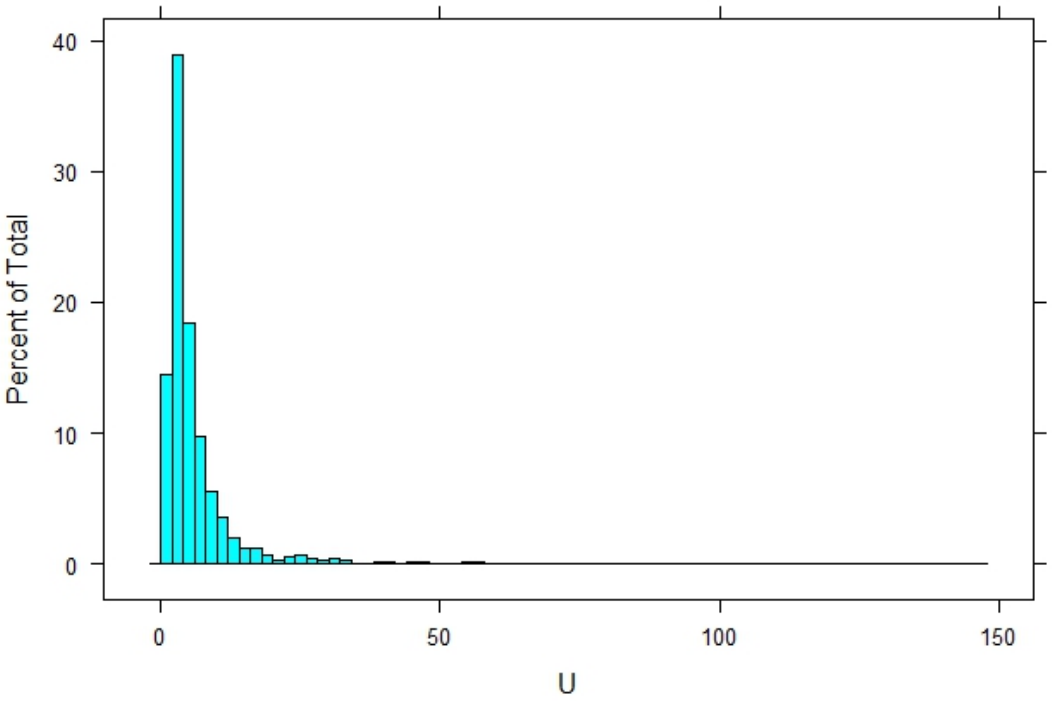
Task 2

After the installation of packages, we visualize the data using a 3D scatterplot.

```
rm(list=ls())

setwd("")

require(lattice)

require(sp)

require(gstat)

require(scatterplot3d)

# Load up the data

data= read.table("t", header = TRUE,dec=",")

head(data) #head(), tail() Return the First or Last Part of an Object

fix(data)   #Fix an Object

attach(data)

coordinates(data) <- c("POINT_X", "POINT_Y")

class(data)

statistic=summary(data)

x<-data$POINT_X

y<-data$POINT_Y
```

```
U<-data$U_DN_PPM
scatterplot3d(x,y,U, color="blue",highlight.3d = T, cex.axis=1.3,xlab="λ(degree)", ylab="ϕ (degree)",
      zlab="observed value", pch=15, box=T, cex.symbols=0.5, cex.lab = 1)
```



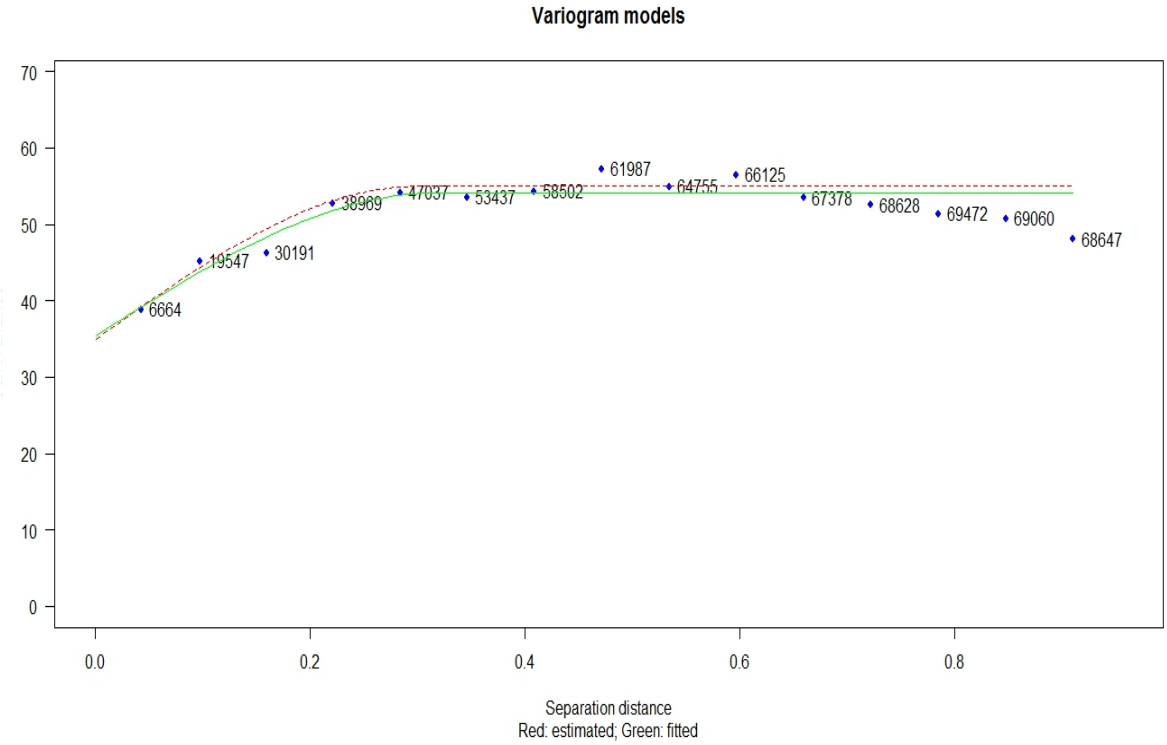


We can see from the scatterplot and histogram that our values for Uranium are mostly around the 1 till 10.

Task 3

We have computed the default empirical variogram of the U values in the calibration dataset. Then we computed the fit.variogram ,which fit ranges and sills from a simple or nested variogram model to a sample variogram.

In the end we plotted the emprical variogram with the fitted variogram models of different classes superimposed, to visualise the effect of the automatic fit.



Code:

```
#########################################################
#Compute the default empirical variogram of the U values in the calibration dataset; and plot it.
#########################################################
#   variogram {gstat}, Calculate Sample or Residual Variogram or Variogram Cloud
v <- variogram(U_DN_PPM ~ 1, loc=data)
print(plot(v, plot.numbers=T, pch=20, col="blue"))
#########################################################
print(show.vgms())
#########################################################
print(show.vgms(models=c("Sph") , sill=20, nugget=35, range=0.3, max=1.6))
vm <- vgm(20,"Sph",0.3,35)
print(vm)
class(vm)
```

```
print(plot(v, plot.numbers=T, pch=20, col="blue", model=vm))
#######################################################
#fit.variogram:Fit ranges and/or sills from a simple or nested variogram model to a sample variogram
vmf <- fit.variogram(v, vm)
print(vm)
print(vmf)
vmf$range - vm$range #subtract of vmf-vm range and sill
vmf$psill - vm$psill
sum(vmf$psill) - sum(vm$psill)
#######################################################
### Plot the empirical variogram with the fitted model superimposed
#######################################################
print(plot(v, plot.numbers=T, pch=20, col="blue", model=vmf))
### What proportion of the total variance in Ur is explained by the fitted variogram model?
#######################################################
1-vmf$psill[1]/sum(vmf$psill)
### Plot emprical variogram with the fitted variogram models of
### different classes superimposed, to visualise the effect of the automatic fit.
#######################################################
plot(v$gamma ~ v$dist, xlim=c(0, max(v$dist)*1.05), ylim=c(0, max(v$gamma)*1.2),
    pch=20, col="blue", cex=1.2, xlab="Separation distance", ylab="Semivariance",
    main="Variogram models",sub="Red: estimated; Green: fitted")
text(v$dist, v$gamma, v$np, pos=4)
lines(variogramLine(vm, maxdist=max(v$dist)), col="red", lty=2)
lines(variogramLine(vmf, maxdist=max(v$dist)), col="green")
```

Task 4

Code:

```
#########################################################
### Fitting anisotropic variograms####
#########################################################
v.a <- variogram(log(U_DN_PPM)~1, data, alpha=c(30,120))
print(plot(v.a, main="Directional Variograms",sub="Azimuth 30N (left), 120N (right)",
        pl=T, pch=20, col="red"))
vm.a<-vgm(020,"Sph",0.3,35, anis=c(30, 0.5))
print(vm.a)
class(vm.a)
print(plot(v.a, plot.numbers=T, pch=20, col="blue", model=vm.a))
vmf.a <- fit.variogram(v.a,vm.a)
```