

Change of support - R practicum

In this practicum we will investigate a dataset studied in [Craigmile et al. \[2006\]](#). Here is their description of the dataset:

“In recent years there has been a great deal of interest in monitoring nitrogen levels in the Chesapeake Bay, the largest estuary in the United States. Parts of six states make up the Chesapeake Bay watershed: Delaware, Maryland, New York, Pennsylvania, Virginia, and West Virginia, as well as the District of Columbia. Increased nitrogen levels increase algal growth. The blanketing effect of algal growth, along with the decomposition of these algae, in turn decreases the amount of dissolved oxygen in the water. As a consequence, increasing nitrogen levels pose a substantial threat to the wildlife and ecology of the Bay and the surrounding watershed. In a recent agreement between the Chesapeake Bay Program partners, every state is mandated to regulate the nitrogen levels (both from tributaries and from the air) in their part of the Bay (Chesapeake Bay Program Partnership, 2000).

We consider a spatial dataset of 49 measurements of the surface-nitrogen concentration (in mg/l) taken in May 1995 throughout the Chesapeake Bay. This same dataset was used in [De Oliveira and Ecker \[2002\]](#) for hotspot detection.”

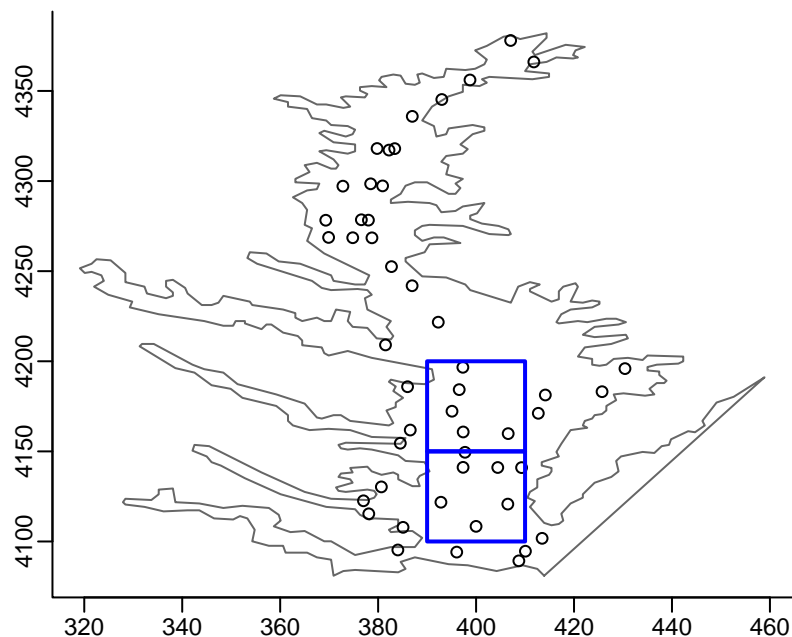


Figure 1: The region of interest and the locations of the nitrogen samples. The two blue regions are the blocks of study.

The set of spatial locations are shown in Figure 1. In environmental monitoring we are often interested in monitoring a quantity summarized over specific regions. Suppose we wish to predict the average log nitrogen concentrations in the two blue areas in the figure. Before we can do this we need to carry out a full geostatistical analysis of the data.

We will use the geoR library [Ribeiro Jr and Diggle, 2001, Diggle and Ribeiro Jr, 2007]:

```
## load the geoR library
library(geoR)
```

We start by reading in the data, storing the dataset as a data frame.

```
# Read in the data, storing as a data frame called 'ches'
ches <- read.table("chesbay.txt", header=TRUE)
```

```
## What are the name of the variables?
names(ches)
```

```
[1] "id"          "northing"    "easting"     "latitude"    "longitude"   "nitrogen"
```

In this analysis we are interested in the coordinates variables (`northing` and `easting`) and the surface nitrogen levels in mg/l (`nitrogen`).

We will also load in a file containing the spatial coordinates of the boundary of Chesapeake Bay. We will use this to produce our maps.

```
## read in a dataset containing the boundaries of Chesapeake Bay.
boundary <- read.table("boundary.txt", header=TRUE)
```

1 Exploratory spatial data analysis

Start by looking at the locations of the nitrogen samples over the map of the Chesapeake Bay.

```
## Show the boundaries in gray.
plot(boundary, xlab="Easting", ylab="Northing", col="gray40", type="l")
```

```
## Plot the locations of the nitrogen samples over the top.
points(ches$easting, ches$northing)
```

1. Describe the sampling arrangement.
2. What is good about the sampling? What is bad about the sampling?

In our example we wish to monitor the average nitrogen level in these two rectangular regions:

```
## Show block 1
rect(390, 4100, 410, 4150, border="blue", lwd=2)
```

```
## Show block 2
rect(390, 4150, 410, 4200, border="blue", lwd=2)
```

Verify your graphics with Figure 1. We next show the nitrogen concentrations on the map. Describe what you see.

```
## Show the boundaries in gray.
plot(boundary, xlab="Easting", ylab="Northing", col="gray40", type="l")

## plot the nitrogen values
text(ches$easting, ches$northing, round(ches$nitrogen,1), cex=0.7)
```

Now carry out an exploratory spatial analysis on the **log nitrogen** values. Make sure that you investigate the following:

1. Demonstrate that there are spatial trends. Predict the trends naively with ordinary least squares using the easting and/or the northing coordinates.
2. After you detrend your data, produce a semivariogram for the residuals using the `variog` R function. Explain what spatial dependence you see.
3. Is the spatial dependence isotropic? (hint: use the `variog4` function). If the variogram is not isotropic, consider a transformation of the coordinate system to make the residuals closer to isotropy.
4. Fit an exponential covariance function using weighted least squares (`variofit`) to the transformed coordinate-system residuals.

Note that the exponential covariance structure is clearly an approximation, but will be sufficient for our analysis.

2 Estimating the spatial model parameters via likelihood methods

Suppose that `tx` are your transformed x coordinates, `ty` are your transformed y coordinates and `trend.resids` are your residuals. We can then create a geodata object for our residuals.

```
## Create a geodata object for the residuals,
Rmy.resids <- cbind(tx, ty, trend.resids)
Rnitro <- as.geodata(Rmy.resids, coords.col=1:2, data.col=3)
```

Now, assuming Gaussianity for the residuals (is this reasonable?), we estimate the parameters using maximum likelihood:

```
## Estimate the covariance parameters using maximum likelihood,
## assuming that the residuals are Gaussian.
ml.fit <- likfit(Rnitro, cov.model="exp", fix.nugget=FALSE,
                nugget=1, ini.cov.pars=c(1,1))
summary(ml.fit)
```

We could also use restricted maximum likelihood:

```
## Estimate the covariance parameters using REML
reml.fit <- likfit(Rnitro, cov.model="exp", fix.nugget=FALSE,
                  lik.method="REML",
                  nugget=1, ini.cov.pars=c(1,1))
summary(reml.fit)
```

Supposing that `emp.var.final` is your final empirical variogram, here is a comparison of the model fits (you will need to rename the R object).

```
## Show the fitted ML-based and REML-based variograms over
## the robust binned estimator.
plot(emp.var.final)
lines(ml.fit, lty=1, col="blue")
lines(reml.fit, lty=2, col="red")
```

How do the two model fits compare?

These model fits are only for the residuals from the ordinary least squares fit to the spatial trend. We can fit the model to the original dataset – this is more involved:

```
## We can also fit a REML-based model to the original data,
## including a linear trend.
```

```
## Fit we set up the new geo.data object
my.data <- cbind(tx, ty, log.nitro)
Onitro <- as.geodata(my.data, coords.col=1:2, data.col=3)
```

```
## You need to fill in your trend model here after the tilde ~
your.trend.model <- ~ **PUT YOUR TREND MODEL HERE**
```

```
## The we estimate the covariance parameters using REML
trend.reml.fit <- likfit(Onitro, cov.model="exp", fix.nugget=FALSE,
                        trend=trend.spatial(your.trend.model),
                        lik.method="REML",
                        nugget=1, ini.cov.pars=c(1,1))
summary(trend.reml.fit)
```

Write down the statistical model that you are fitting to the log surface nitrogen concentrations, and describe these model summaries.

3 Spatial prediction (Kriging)

In this section, we will demonstrate predictions of the residual field (without the trend component). We first need to set up a grid of locations that we will predict at.

```
## What locations shall we predict at?
px <- seq(360, 440, length=30)
py <- seq(4080, 4390, length=100)

pred.sites <- expand.grid(px, py)
```

We can look at where these sites are located.

```
## Show these prediction locations on a map.
par(mfrow=c(1,1))
plot(boundary, xlab="Easting", ylab="Northing", col="gray40", type="l")
points(pred.sites, pch=4, cex=0.5)
```

Is there something strange about the choice of predicted locations? (How might you change the choice of predicted locations?)

We need to make sure that we rescale these predicted locations, before we get `geoR` to produce our predictions (to match with the spatial coordinate system in our model for covariances).

```
## Define your transformed x and y coordinates (fill in the blanks
## by transforming pred.sites[,1] and pred.sites[,2]).
ptx <-
pty <-

## To kriging we have to remember to rescale the locations
predicted.sites <- cbind(ptx, pty)
```

After rescaling, we can calculate the ordinary kriging predictor and its standard error (SE) using the function `krige.conv` (`conv` stands for “conventional”).

```
## Let us start by predicting the residual field
krige <- krige.conv(Rnitro, loc=predicted.sites,
                   krige = krige.control(
                     cov.model=reml.fit$cov.model,
                     cov.pars=reml.fit$cov.pars,
                     nugget=reml.fit$nugget))
```

Here is some code to explore these predictions graphically.

```

## Now produce some pretty pictures of the predictions.
par(mfrow=c(2,1))

## Plot the axes
plot(boundary, xlab="Easting", ylab="Northing", type="n", main="Prediction")

## Show the predicted values, along with a contour map to indicate the values
image(px, py, matrix(krige$pred, length(px)), add=T)
contour(px, py, matrix(krige$pred, length(px)), add=T, nlevels=10)

## Add the boundaries
lines(boundary, col="gray40")

## And the locations of the nitrogen samples over the top.
points(ches$easting, ches$northing)

## Show the boundaries in gray.
plot(boundary, xlab="Easting", ylab="Northing", type="n", main="SE of Prediction")

## Show the SE of the predicted values, along with a contour map to
## indicate the values
image(px, py, matrix(sqrt(krige$krige.var), length(px)), add=T)
contour(px, py, matrix(krige$krige.var, length(px)), add=T, nlevels=10)

## Add the boundaries
lines(boundary, col="gray40")

## Plot the locations of the nitrogen samples over the top.
points(ches$easting, ches$northing)

```

Describe the predicted field of values. Where do we predict that the residual log nitrogen process has higher values? Where do we predict it has lower values? Describe the standard errors of your predictions – what spatial patterns do you see in the uncertainty?

if you have time, investigate how you might include the trend component in your prediction (Consider the following R code).

```

## define the trend for the predictions using the columns
## of predicted.sites
your.predicted.trend.model <- ~ ** FILL IN HERE **

```

```
## Kriging, with a trend
krige <- krige.conv(Onitro, loc=predicted.sites,
                  krige = krige.control(
                    type.krige = "OK",
                    trend.d = trend.spatial(your.trend.model),
                    trend.l = trend.spatial(your.predicted.trend.model),
                    beta=trend.reml.fit$beta,
                    cov.model=trend.reml.fit$cov.model,
                    cov.pars=trend.reml.fit$cov.pars,
                    nugget=trend.reml.fit$nugget))
```

4 Predicting blocks – an example of change of support

We are now in a position to predict at a different spatial scale – the block level. Read pages 14–19 of the lecture notes.

1. Try to predict the average log nitrogen level in the two blocks defined in Figure 1. The coordinates are:
 - (a) 390–410 E, 4100–4150 N and
 - (b) 390–410 E, 4150–4200 N.

If you have less time, only consider predicting the detrended log nitrogen level.

2. Produce standard errors for your predictions and try to estimate the covariance between the two blocks.

References

- P. F. Craigmile, N. Cressie, T. J. Santner, and Y. Rao. A loss function approach to identifying environmental exceedances. *Extremes*, 8(3):143–159, 2006. URL <http://dx.doi.org/10.1007/s10687-006-7964-y>.
- V. De Oliveira and M. Ecker. Bayesian hot spot detection in the presence of a spatial trend: application to total nitrogen concentration in Chesapeake Bay. *Environmetrics*, 13:85–101, 2002.
- P. J. Diggle and P. J. Ribeiro Jr. *Model Based Geostatistics*. Springer, New York, 2007.
- P. J. Ribeiro Jr and P. J. Diggle. geoR: a package for geostatistical analysis. *R-NEWS*, 1: 14–18, June 2001. URL <http://CRAN.R-project.org/doc/Rnews/>. ISSN 1609-3631.